AFIT/GCS/ENG/97D-03

# AN INVESTIGATION INTO THE USE OF SOFTWARE PRODUCT METRICS FOR COBOL SYSTEMS

THESIS

Richard Edward Boone, First Lieutenant, USAF

AFIT/GCS/ENG/97D-03



Approved for public release; distribution unlimited

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

AFIT/GCS/ENG/97D-3

# AN INVESTIGATION INTO THE USE OF SOFTWARE PRODUCT METRICS FOR COBOL SYSTEMS

THESIS

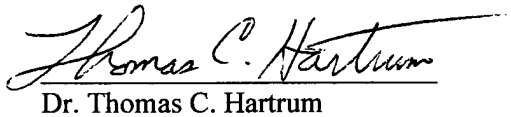
1Lt Richard E. Boone

Presented to the Faculty of the Graduate School of Engineering

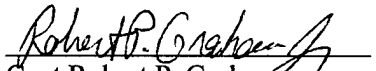
of the Air Force Institute of Technology

in Partial Fulfillment of the Requirements for the

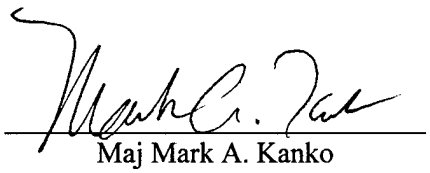
Degree of Master of Science in Computer Systems



Dr. Thomas C. Hartrum
Member



Capt Robert P. Graham
Member



Maj Mark A. Kanko
Chairman

## *Acknowledgments*

This research could not have been accomplished without the help, patience, and understanding of many people. Being a novice in the software engineering field coupled with that fact that I'd been out of school for four years while working in a non-computer related job, I welcomed all the help I could get.

First, and foremost, I want to thank my wife for giving me words of encouragement and giving me the confidence I sometimes lacked. She never doubted my abilities even when I sometimes doubted them myself. Thanks for being a great wife as well as my best friend.

I'd like to also thank Maj (Dr.) Mark Kanko for guiding me in an unfamiliar area. Without him, I would have been lost. He was always there to answer questions, give direction, and supply encouragement. I'd especially like to thank him for his effort in finding a system suitable for this study. We ran into a lot of red tape, roadblocks, and dead-ends before he finally located a system I could use.

I'd also like to thank the other members of my thesis committee: Dr. Thomas Hartrum and Capt. (Dr.) Robert Graham. Their expertise in software engineering was invaluable. I would like to especially thank Dr. Hartrum for going out of his way to supply the software tool I needed for this research as well the funding I needed for travel.

I also owe a great debt of gratitude to two AFIT Ph.D. students (one has since graduated): Maj (Dr.) Rick Sward and Maj Thomas Schorsch. They both took time out of their busy schedules to help me with the *Refine*™ code I used to conduct my analysis.

I'd also like to thank my fellow GCS classmates and other "AFITeers" who participated in our once-a-week afternoon basketball rendezvous. The games proved to be a great channel to vent the stressors of AFIT life!

Finally, I want to thank the people at AMC/CSS for their tremendous support in providing me with the data I needed. I'd like to thank Maj McCanne for allowing me to use his system in this research. I truly owe a great deal of thanks to SSgt Brian "Ski" Andrzejewski and his team. Ski and his team put in many hours ensuring I had the data I needed. He went out of his way to answer questions and provide follow-up information throughout my entire research effort. Thanks a million.

Rick Boone

*Table of Contents*

## *List of Figures and Tables*

## *Abstract*

This thesis investigated several hypotheses that specific product measures could be used to predict later software lifecycle process or product measures. It collected software product and process measures from four consecutive major releases of a large COBOL legacy system (400K LOC). It used an automated tool to extract the product measures directly from the source code for each release. Several programs were written using this tool to calculate the product measures for each procedure. The types of product measures extracted were size and specific complexity measures (cyclomatic complexity, information flow, nesting level).

A statistical software package was used to calculate sample correlation coefficients between the product measures and the maintenance process measures provided by the owners of the COBOL software system. A 95% confidence interval was computed for each sample correlation coefficient that showed a strong or moderate linear correlation. The maintenance process measures provided were manhours used for each program changed or added, and defects detected during each change request. Sample correlation coefficients were derived to see if product measures such as size and cyclomatic complexity could reveal trends that could be used to estimate other software lifecycle measures such as effort and defects.

The hypotheses to this research could neither be accepted nor rejected because the process measures collected by the system's owners were recorded at a level too high for statistical analysis. The weaknesses are identified in the way these measures are collected, and suggestions are provided on how measures can be better identified and recorded.

# 1. Introduction

The focus of this research was to use specific source code product measures of a large COBOL legacy system to predict other lifecycle activities. Size and complexity measures were gathered from the source code to be used as product metrics. The product metrics used are defined in Chapter 2 of this thesis. The research analyzed four consecutive versions or "releases" of the COBOL system along with specific process measurements provided by the owners of the system. The process measures provided were defect data and manhour data. Once the product measures were gathered, the research investigated possible correlations between the product measures and the process measurements for each version. This was done to see if the product measures collected can be used as predictors for lifecycle process measures or activities. This research also investigated possible correlations between different types of product measures to see if product measures can be used to predict other lifecycle product measures.

This chapter will explain some background on why a COBOL system was chosen for the study as well as how measurement can help software engineers control processes. It introduces the problem statement, outlines the scope of the thesis, and presents some hypotheses. Finally, it describes the document layout.

## 1.1 Background

This section provides some background on the COBOL programming language, software maintenance, and the need for measurement in software engineering. It also introduces the software system being analyzed and provides some background about the owning organization.

### *1.1.1 The COBOL Programming Language*

The COBOL programming language continues to be the most widely used and maintained programming language in the Department of Defense and many commercial industries. A 1994 study conducted by International Data Corporation found that 60 percent of companies worldwide currently using COBOL plan to maintain, or actually increase, their reliance on COBOL in the future [Tri95]. The Air Force alone has organizations that maintain multi-million-line COBOL programs. Most of these systems are old and are considered "legacy" systems. There is no exact definition of when code is considered to be "legacy" code. In general, a system that has been in the field for several years and has gone through several revisions is considered to be legacy code.

"COBOL" stands for Common Business Oriented Language. COBOL was one of the earliest high-level languages and was designed for business-oriented problems. Unlike other languages, COBOL programs are organized into four different *divisions*. These are the *Identification Division*, the *Environment Division*, the *Data Division*, and the *Procedure Division*. The *Identification Division* is where the program name, author, date of creation, etc. are entered. The *Environment Division* describes the programs system environment such as hardware and operating system. The *Data Division* describes, declares, and initializes all the variables and files that will be used in the program. This is the only place variables can be declared, hence only global variables can exist in a COBOL program. The *Procedure Division* contains the executable statements of the program. It consists of a series of COBOL paragraphs, or procedures [Gra94]. This research will consider a COBOL procedure to be a module. The terms paragraph, procedure and module will be used interchangeably throughout this document. The COBOL *Procedure Division* is flexible because it can be written with a driver procedure that

controls the execution of each procedure, or the code can be written so each procedure is executed sequentially without a driver, sometimes termed "drop through". Because all variables are global in COBOL, procedures do not pass parameters between each other [Nic86].

Only a limited amount of product metrics work has been done with the COBOL programming language. Most of the software metrics community has overlooked COBOL and focused their sights on newer languages like Ada or C++. Perhaps this is because of the perception that COBOL is now predominantly a maintenance language and no longer a development language.

Unlike languages such as Ada and C++, COBOL is not designed for object oriented programming. Rumbaugh et al. state that a language is object oriented if it includes the following four aspects [Rum91]:

1. Identity - data is grouped into discrete, distinguishable entities called objects,
2. Classification - objects with the same attributes and operations are grouped together,
3. Polymorphism - the same operation may behave differently on different data, and
4. Inheritance - sharing of attributes and operations based on a hierarchical relationship.

Although COBOL supports identity and classification, it does not support polymorphism or inheritance. These disadvantages, coupled with the fact that all variables are global, make it difficult for a COBOL programmer to encapsulate data, hide information, and modularize procedures.

With the enormous amount of COBOL code maintained by the Air Force, it is difficult to ensure good-quality programs are being written and maintained using only manual methods.

Using metrics to gauge activities is a practice that can help software managers and developers produce quality software. This research will look at whether software product metrics could be useful to managers and developers of COBOL legacy systems by providing them an additional tool to control and estimate lifecycle activities. The ability to control and estimate lifecycle activities could help software managers plan a more accurate schedule and reduce development costs.

### *1.1.2 The Cost of Software Maintenance*

Technological advances in the past two decades have shifted the costs of hardware and software in opposite directions. In the late 1970s, hardware accounted for 80% and software for 20% of the military's computing cost. With advancing technology, hardware has become easier to manufacture. Hence, the cost of hardware has dropped drastically and now consumes only 20% of computing costs. In contrast, advances in technology have boosted the demand for complex software in the military. Due to the increase in demand for software, organizations employing thousands of people have been established worldwide just to develop and maintain software. Thus, software now accounts for 80% of the military's computing cost [Con86].

With the continuing reduction of the Department of Defense budget, military organizations have been asked to reduce their annual spending drastically. In software organizations throughout the DoD, much of the spending is dedicated to maintenance of source code. Improving the maintenance process of Air Force software organizations could save the government millions of dollars annually. Applying metrics to software development processes could help identify problem areas that, once rectified, could save time and money on software projects.

### *1.1.3 The Need for Measurement in Software Engineering*

Measurement is important for three basic activities. Fenton claims measurement can [Fen97]:

1. help us understand what is happening during development and maintenance,
2. allow us to control what is happening on our projects, and
3. encourage us to improve our processes and products.

Measurement in the software engineering realm has historically been neglected. Although measurement goes hand-in-hand with science and engineering, measurement has always been considered a luxury in the software engineering arena. For most development projects [Fen97]:

1. Managers fail to set measurable targets for their software products. For example, they promise that the product will be user-friendly, reliable, and maintainable without specifying clearly and objectively what these terms mean or how they will be measured. As a result, when the project is complete, they cannot tell if they have met their goals.

2. Managers fail to understand and quantify the component costs of software projects. For example, most projects cannot differentiate the cost of design from the cost of coding or testing. Since excessive cost is a frequent complaint from many of their customers, managers cannot hope to control costs if they are not measuring the relative components of cost.

3. Managers do not quantify or predict the quality of the products they produce. Thus, they cannot tell a potential user how reliable a product will be in terms of likelihood of failure in a given period of use, or how much work will be needed to port the product to a different machine environment.

4. Managers allow anecdotal evidence to convince them to try yet another revolutionary new development technology, without doing a carefully-controlled study to determine if the new technology is efficient and effective in their specific environment.

When measurements are made, they are not done as often as they should be, they are not taken consistently, and they are rarely complete [Fen97]. To ensure consistency and accuracy in collecting metrics, it is important that processes be established that instruct software developers

and maintainers on what to measure and how to measure it accurately. Successful metrics programs are highly dependent on collecting relevant, accurate data [Kan93].

### *1.1.4 Processes and the Capability Maturity Model (CMM)*

In November 1986, the Software Engineering Institute (SEI) began developing a process maturity framework designed to help organizations improve their software process. The ultimate product was the CMM [Pau93]. The CMM gives guidance on how to control processes for developing and maintaining software. It also gives direction on how an organization can evolve to disciplined software engineering practices. This model assigns a *maturity level* from one to five to an organization, based on how well processes are defined and used. Figure 1-1 gives a graphical depiction of the CMM. The levels are designed so that the lower level capabilities act as foundations to build on to progress to the next higher level of the CMM [Hum92].

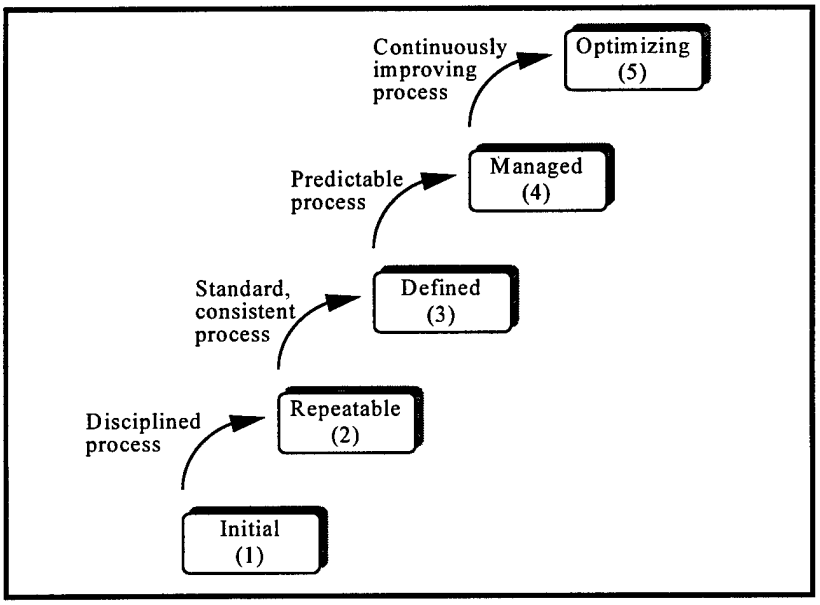



**Figure 1-1.** The Five Levels of Software Process Maturity

The need for defined processes is crucial with the continuing demand for software organizations to achieve a higher level of the CMM. Defined processes are needed for activities that are performed repetitively. Defined processes help people:

1. plan and track their work,
2. by guiding them through an activity, and
3. evaluate and improve the way an activity is done.

Humphrey [Hum89] states that organizations must perform the following six steps to improve their software development and maintenance capabilities:

1. understand the current processes,
2. develop goals for desired processes,
3. create a prioritized list of required process improvement actions,
4. make a plan to accomplish these actions,
5. commit the resources to execute the plan, then
6. start over at step 1.

One particular area where defined processes are needed before an organization can move to the highest level of the CMM is metrics collection and use. Organizations with successful measurement programs report the following benefits [Bau92]:

1. insight into product development,
2. ability to quantify tradeoff decisions,
3. better planning, control, and monitoring of projects,
4. better understanding of both the software development process and environment
5. ability to identify areas of potential process improvement as well as an objective measure of the improvement efforts, and
6. improved communication.

Measurement is important in software engineering because developers can measure characteristics of the software to determine if requirements are complete, if the design is of high quality, or to see if the system is ready to be tested. Project managers can use measurement to help them determine when software will be ready for delivery and whether the budget will be met or exceeded [Fen97].

### *1.1.5 The HOST Software System*

The Air Mobility Command (AMC) / Computer Systems Squadron (CSS) is one of several organizations that maintain COBOL legacy code in the Air Force. This research analyzed a system maintained by AMC/CSS that has been in the field for over 20 years and used by every Department of Defense (DoD) organization worldwide that ships or receives air cargo. This system, called the Headquarters On-Line System for Transportation (HOST), is comprised of eight separate subsystems that provide authority to move, visibility, accountability, tracking and history data for AMC cargo shipments and commercial movement. The HOST system's functions include cargo processing, data updating, reports output, cargo movement, data inquiry, historical data retention, and system maintenance [Pin97].

AMC/CSS, in an effort to improve processes and attain a higher CMM level certification, collect and use process metrics on their software. Collecting and using process metrics is meant to help them continually improve their process of maintaining all of their software products.

### *1.1.6 Types of Software Metrics*

Conte et al. [Con86] state that software metrics are often classified as either process metrics or product metrics. Process metrics, which AMC/CSS maintain, are used to measure either development or maintenance processes. Process metrics quantify attributes of the development or maintenance processes and environment. Process metrics include measures such as the experience of programmers, man-hours for coding, and cost of development and maintenance. Product metrics apply to software products such as a program's source code, system specifications, or program documentation. Product metrics, however, are measures of what is being produced by the process. Product metrics measure attributes like the size of the

program, the logic structure complexity, and the function of the product [Con86]. Arguments have also been voiced for a third class of metrics, resource metrics, that focus on any input to a process, for example programmers or hardware [Fen97]. AMC/CSS currently collects process metrics but has no activity that collects or uses product metrics.

Product metrics, like Halstead's Software Science [Hal77] and McCabe's Cyclomatic Complexity [McC76], have existed for over 20 years and are used widely in commercial industries in conjunction with process metrics. These metrics have been proven to provide more accurate assessments of software when used along with process metrics [She88]. For example, McCabe has shown that any module with high cyclomatic complexity correlates to a high number of faults experienced by the system [McC96].

This thesis research collected and analyzed software product measures on a specific COBOL legacy system. It attempted to correlate these product measures with specific process measures. This was done to see if these specific product measures could be used as predictors of specific process measures or activities. This research also attempted to correlate the product measures among themselves. This was done to see if product measures could be used to estimate other product measures.

AMC/CSS, as well as other software organizations, can use the results of this thesis to implement a metrics process of collecting and using product measures to estimate or predict lifecycle activities in the development or maintenance of their software.

### *1.2 Problem Statement*

Currently, AMC/CSS neither collects nor uses product metrics to analyze COBOL legacy programs they maintain. They rely solely on process metrics to evaluate their software. This research investigated whether product measures derived from source code could be used as

estimations tools to predict other measures in the software's lifecycle. The ability to predict future activities gives managers and developers the capability to plan for these activities. This could save time and money as well as help software managers provide the customers with a more accurate estimation of the software project's completion.

### *1.3 Scope*

This research collected software product measures from the HOST COBOL legacy system. It used these product measures to attempt to draw correlations with process measures currently being maintained for the system. This research investigated whether or not product metrics, used early in a software system's lifecycle, could be used to predict later software lifecycle activities. In addition, the results will be used to advise AMC/CSS on how they could use product metrics to enhance their software metrics program.

The product measures gathered were McCabe's Cyclomatic Complexity number [McC76], maximum nesting level, Henry and Kafura's information flow measure [Hen81], and size measures such as number of statements in a module and number of modules in a program. Descriptions of these metrics are provided in Chapter 2 of this document.

### *1.4 Hypotheses*

One area this research investigated was the degree of linear correlation between the product measures collected and the process measures provided by the owner of the system. The degree of correlation was measured to see whether the product metrics derived from the source

code could be used as predictors to the process measures such as defects and manhours. This research hypothesized that:

1. the number of manhours needed to write or change code is affected by the complexity of the code,
2. the number of changes in the code (added, deleted, or changed) will correlate with the number of manhours needed, and
3. the number of defects found would correlate with modules having high complexity.

This research also investigated the degree of linear correlation among the product measure collected. McCabe [McC76] claims that size and cyclomatic complexity measures are independent of one another; however, Shepperd [She88] and Fenton [Fen97] claim the contrary. This research investigated the degree of linear correlation between these two measures as well as the other product measures collected. It also looked at how increases or decrease in one product measure from revision to revision may correlate with increases or decreases in other product measures.

### *1.5 Document Overview*

The next chapter reviews the literature that provided background information on software measurement and source code product metrics. It specifically describes the product measures that were collected for this study as well as how the product measures have been used to improve processes. Chapter 3 describes the thesis methodology. Chapter 4 reports the data analysis and results. Finally, Chapter 5 provides conclusions, offers recommendations for process improvement, and gives suggestions for further study.

# 2. *Literature Review*

This chapter reviews the current literature and discusses how software measures are collected and used for both military and commercial practices. It presents some previous metrics work done on COBOL systems. It describes the usefulness of measurement in the software community. It gives a definition of software product metrics and describes the different types of software product metrics this research used. It shows that these metrics have been established and recognized as valid and useful metrics in the software community. This chapter also explains how these software product metrics are used to evaluate software and estimate future software projects.

## *2.1 Previous COBOL Metrics Research*

Little work has been done with product metrics for the COBOL programming language. Gibson and Senn [Gib89] conducted experiments that used complexity metrics from COBOL code to investigate the relationship between system structure and maintainability. One of the objectives of their research was to see if cyclomatic complexity decreased whenever software "improvement" work was done on a system. Their findings revealed that cyclomatic complexity did decrease slightly when "improvement" work was done on a system.

Moe [Moe91] did some research collecting different types of complexity metrics from COBOL code using a lexical analyzer. The complexity measures she investigated included McCabe's Cyclomatic Complexity; however, she did not use Henry and Kafura's Information Flow measure in her research. She investigated whether different types of complexity metrics derived from source code gave relatively the same results. That is, if a specific segment of code

had high complexity for one type of measurement, would the same code have a high complexity for a different type of measurement. She discovered a correlation between the types of complexity measures for the particular system she analyzed.

Although some work has been done with product metrics and COBOL, this research found no work that attempted to investigate relationships between product measures and other lifecycle measures for COBOL systems. This is quite remarkable considering COBOL is still the most widely maintained programming language in the DoD and the world [Tri95].

## *2.2 Using Measurement for Estimation*

Why should we be interested in software metrics? The main reason is to control what we're doing. Fenton cites DeMarco's rule: "You can neither predict nor control what you cannot measure" [Fen97]. If one is to talk about the quality of software one must have a means of measuring it. Quantitative measurement is necessary to assess a certain procedure or tool that is being used to improve a product or a process [Fen97]. Despite the simplicity of the idea of trying to impose order upon software engineering and engineers via measurement, progress in software metrics has been very limited [She92].

The ability to predict gives one the ability to plan. Poor planning by managers has been cited as the main reason development systems go over budget or over schedule. If managers have tools that enable them to accurately predict attributes of a software system, managers will be better equipped to plan the development of that software system.

Valid and accurate measurements help software engineers accurately plan and estimate their projects. An effective software measurement system adds value to every phase of the software lifecycle. Also, measures recorded during software development can be used for future projects [Jon91]. Jones continues by stating, "There is a perfect correlation between

measurement accuracy and estimation accuracy: Organizations that measure well can estimate well; organizations that do not measure cannot estimate either" [Jon91].

Fenton discusses four scenarios for using metrics to predict software system attributes [Fen97]:

Scenario 1: Using early product metrics such as size of specification to predict later product metrics such as size of final source code,

Scenario 2: Using process metrics such as average education level of programmers to predict product metrics such as the logic structure complexity,

Scenario 3: Using product metrics such as size of code to predict process metrics such as time needed to code, or

Scenario 4: Using early process metrics such as number of failures during module testing to predict later process metrics such as number of failures during integration testing.

This research focused on measures that fit best into Fenton's Scenario 1 and Scenario 3 descriptions. It investigated whether product measures, or in this case, COBOL source code measures could be used to predict defect and manhour data. It also investigated whether source code measures could be used to predict other source code measures.

### *2.2.1 Data Essential for Software Measurement*

Jones [Jon91] describes three types of data that must be collected when dealing with software measurement:

1. hard data,
2. soft data, and
3. normalized data.

Hard data deals with objects or attributes that can be quantified with little or no subjectivity. This includes measures such as number of staff assigned to a project, time spent on a project, and the number of defects found and reported. Soft data is more subjective. Examples include

experience of staff, satisfaction of customers, and other environmental factors. Normalized data means that data should be observed relative to other projects. For instance, if a project written in Ada is compared against a project written in Assembly, measures such as lines of code (LOC) must be normalized because it takes approximately four LOC of Assembly to equal one LOC of Ada.

This research focuses on hard data. Although it is suggested that hard data can be measured very accurately, most organizations record their hard data inaccurately [Jon91]. One problem encountered with collecting data is lack of granularity. Jones [Jon91] states that organizations tend to accumulate only cumulative or composite data instead of separating the data into meaningful subsets, such as the effort for requirements, design, and coding. Such high-level data is of little value as there is no true way to validate it. Although the defense industry takes estimation seriously, most defense organizations have spotty and incomplete quality measurements [Jon91]. The hard data measures provided by AMC/CSS and used in this research were defect data and manhour data.

### *2.2.2 Defect Data*

Jones [Jon91] states that since the cost of finding and fixing bugs has historically been the largest factor in software cost, one of the most important hard-data measurements is defect count. A defect can be a requirements error, design defect, coding defect, documentation defect, or a bad fix. A good defect count measurement system can improve an organization's defect removal efficiency. The overall U.S. average of defect removal efficiency by organizations is only about 75% [Jon91]. Measuring defect removal efficiency is a sign that an organization is on the leading edge. Organizations with a high defect removal efficiency (95% or higher) tend

to be optimal in three other aspects when compared to other organizations that have projects of the same size and type [Jon91]:

1. they have the shortest schedules,
2. they have the lowest quantity of effort level in terms of person-months and person-hours, and
3. they have the highest levels of user satisfaction after release.

The average organization routinely delivers a software system with up to 25% of the total defects still undiscovered, while leading-edge organizations can reduce this rate to 5% [Jon91].

### *2.2.3 Manhour Data*

Manhour effort is sometimes difficult to quantify. All too often software managers over-estimate their team's ability to produce code. Instead of using empirical data to estimate production, they rely on educated guesses. Some of the reasons these estimates can be so far off is that managers fail to account for time loss due to machine downtime, meetings, paperwork, personal time, and sickness, etc. [Bro95]. One must attempt to take these factors, and others, into account if manhour effort is to be recorded accurately.

Collecting manhour data during project development is intuitively useful for predicting other development projects of the same type. However, using manhours as a predictor of the amount of code that can be written can be misleading. Manhour data collected while developing relatively simple code, with little complexity, could reveal relatively high productivity. However, manhour data collected while developing highly complex code, could suggest low productivity. For these reasons, the complexity of the code must be factored in when estimating the manhours needed to write the code.

## 2.3 *Software Product Metrics*

The concept of software product metrics is not new. Over 20 years ago Halstead originally proposed a family of software measures known as Software Science [Hal77]. Halstead tried to capture aspects of a program that were similar to physical and psychological measurements in other disciplines. He claimed his software science metric could be used to predict the length of a program and effort level needed just by observing certain attributes of the program's code [Hal77]. Although Halstead's metric has been questioned repeatedly by software engineers and statisticians, his effort established the new discipline of software science known as software product metrics. The software community now has a variety of software product measures that can be used to analyze source code.

Two particular types of product measures that dominate the interests of software engineers are program size and various types of program complexity measures. This research effort focused on these two types of product metrics. The following introduces and defines the software product measures used in this research.

### 2.3.1 *Program Size Measure*

There are many ways to measure the size of a program. One can count source lines of code (LOC), the number of modules, function points, or the number of statements, etc. The LOC measure is the most popular means to measure the size of a program possibly because it is the simplest product measure to collect. The basis for LOC as a valuable measure is that program length can be used as a predictor of program characteristics such as reliability and ease of maintenance [She88]. Since the software community agrees that, to a large extent, the amount of effort necessary to construct a program depends upon the number of lines that are written, the size measure naturally becomes a dominating factor in effort-related studies. The size of a

program is claimed to be an important measure for primarily these reasons: it is easy to compute after the program is completed, it is the most important factor for many models of software development, and productivity is normally based on a size measure [Con86]. It has been suggested that the LOC measure be regarded as a baseline metric to which all other metrics be compared [She88].

The software community has used several different methods to count the LOC measure of a program. Some organizations include comment lines or blank lines when counting the overall LOC measure. However; Conte et al. [Con86] state a line of code is any line of program text that is not a comment or blank line, regardless of the number of statements or fragments of statements on the line. This specifically includes all lines containing program headers, declarations, and executable and non-executable statements. It does not include blank lines or comment lines [Con86]. The important factor to remember when using any size measure is that the method of counting must be consistent when comparing programs against one another, especially within a single organization.

### *2.3.2 Program Complexity and Associated Metrics*

The complexity of a program is of major importance when dealing with product metrics. Wallace et al. [Wal96] state that highly complex software is prone to have more errors than software that is not as complex. Complexity beyond a certain limit hampers a person's ability to understand and interpret information. The same psychological factors that limit a person's ability to do mental manipulations of more than "7 +/- 2" objects also apply to software [Wal96]. Therefore, the more complex a piece of software is when it's time to modify it, the more likely the programmer will have difficulty understanding it. This increases the chances of the programmer making an error in the modification.

Conte et al. [Con86] state that when dealing only with software, complexity is a characteristic of the software interface which influences the resources another system will expend or commit while interacting with the software. However, from the human point of view, complexity deals with aspects of the software that effect the programmer performance in designing, developing, understanding, testing, and maintaining the software. This definition implies that complexity can be attributed to the software itself, its interactions with other systems, or how difficult the code is to read [Con86]. Fenton states that there are four different ways to interpret complexity:

1. problem or computational complexity: Deals with how complex the problem is we're trying to solve,
2. algorithmic complexity: Deals with the complexity of the algorithm used to solve the problem,
3. structural complexity: Deals with the complexity of the structure used to implement the above algorithm, and
4. cognitive complexity: Deals with the effort needed to understand the software [Fen97].

These types of complexities can interact with each other by cause and effect. Reducing or increasing one type of complexity may cause other types of complexity to reduce or increase. A program's modules or procedures can be described as having high or low levels of any combination of the above types of complexity.

Brooks [Bro95] believes that complexity causes difficulty in communication among team members, which in turn leads to product flaws, cost overruns, and schedule delays. Complexity makes programs hard to use, makes programs difficult to extend without causing side effects, and makes programs vulnerable to security trapdoors. Complexity also makes programs harder to understand, therefore making them less reliable [Bro95].

A primary goal of quantifying program complexity is to determine if highly-complex modules could possibly be redesigned to make the module more testable and maintainable therefore increasing its reliability [Moe91]. The ability to measure implies:

1. we can identify high-complexity modules during the design or coding phase (but before integration) and redesign them, and

2. we can identify modules in existing software systems that are potential problems, since complexity has been found to be directly related to error density.

There have been several product metrics developed to quantify a program's complexity. This research involved three such metrics: McCabe's Cyclomatic Complexity, Nesting Level, and Henry and Kafura's Information Flow Measure.

#### *2.3.2.1 McCabe's Cyclomatic Complexity*

McCabe's Cyclomatic Complexity was first introduced in 1976 [McC76]. This measurement focuses on the control structure of a computer program by determining all the possible independent paths from start to finish that can be taken during execution. This metric has remained popular for several years primarily because it is simple to compute.

One may see a problem with this measurement when faced with a backward branch. Backward branches can lead to an infinite number of paths through a program, resulting in an infinite complexity measure. Note, however, that this metric is designed to compute the program's independent paths and not all possible paths, hence, it always produces a finite measure.

McCabe's measure is also known as the cyclomatic number, V(G), which is computed by depicting a program as a strongly connected graph $G$ with $n$ vertices or nodes, $e$ edges, and $p$ connected components. For example, Figure 2-1 is a program control graph with entry node $a$

and exit node *f*. It depicts a program segment that has decision points at *a* and *c*. That means, upon reaching these nodes in the execution of the segment, there are two possible paths the program can take.

The set of linear independent paths of the flow diagram depicted in Figure 2-1 are illustrated in Figure 2-2. A linear independent path is a unique possible path a program can take during execution. For instance, the path *abf* in Figure 2-2 shows one possible path the program segment can take during execution. Paths *acdf* and *acef* represent the only other possible unique paths the program segment can take.

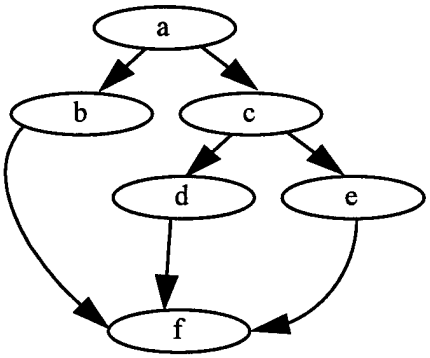



**Figure 2-1.** A flow graph

{abf, acdf, acef}

**Figure 2-2.** Set of linear independent paths

Equation 2-1 computes the cyclomatic number of a program module $G$ which is equal to the maximum number of linearly independent paths:

$$V(G) = e - n + 2p \qquad (2\text{-}1)$$

Computing the cyclomatic complexity for the flow graph depicted in Figure 2-1, $n$ is 6, $e$ is 7, and $p$ is 1(only one component). Using equation 2-1 gives:

$$V(G) = 7 - 6 + 2 = 3$$

McCabe discovered that there is a direct relationship between the associated cyclomatic complexity number of a directed graph and the number of predicates within a program. That is, the cyclomatic number is one more than the number of binary predicates in the statement. A binary predicate is when only two possible conditions can occur at a decision point. For example in Figure 2-3 there are three binary predicate statements (If-Statements) each of which have only two possible conditions. The possible conditions for the first If-Statement are $A > B$ or $A$ not $> B$. Thus, the cyclomatic complexity number of the segment of code in Figure 2-3 is 3. This relationship makes it much easier to compute the cyclomatic complexity number, alleviating the need to construct a directed graph. The cyclomatic number can therefore be conveniently calculated directly from a program's source code [Moe91].

McCabe also noted that compound predicates add to the cyclomatic complexity of a statement. For example in Figure 2-4, the If-Statement predicate is compound because it is checking two separate binary predicates. Because of this, each conjunction in a predicate adds one to the cyclomatic complexity number of a statement. Therefore, the cyclomatic complexity number for figure 2-4 is also 3.

```
IF A > B THEN
  IF B > C THEN
    MOVE 3 TO B
   ELSE
    MOVE 7 TO B
ELSE
  MOVE 5 TO B
END IF.
```

**Figure 2-3.** Multiple If-Statement

```
IF A > B AND B > C
  MOVE 10 TO A
ELSE
  MOVE 5 TO A.
```

**Figure 2-4.** A Compound If-Statement

To compute the cyclomatic number for an entire program $X$ that consists of several modules, the cyclomatic number, $V(X)$, is equal to the sum of the individual complexities of each module. This is due to each module representing a single connected component with one entry and one exit point. For example, let there exist a program $X$ with $p$ connected components, or modules. Let $p(i)$ denote the $p$ unique connected components, where $i$ ranges from 1 to $p$. Let $e_i$ denote the number of edges in the $i^{th}$ connected component, and $n_i$ denotes the number of nodes in the $i^{th}$ connected component.

Then:

$$V(X) = e - n + 2p$$

$$= (\sum_{i=1}^{p} e_i - \sum_{i=1}^{p} n_i) + 2p$$

$$= \sum_{i=1}^{p} (e_i - n_i + 2) \qquad (2\text{-}2)$$

The last sum represents the total cyclomatic number for the program where $p$ is equal to the number of connected components [McC76].

McCabe also points out that the overall cyclomatic number of a program can be easily computed by just counting the number of total binary predicates in the program, including any additions for compound predicates, and adding one for each connected component (module) in the program to the total number [McC76]. For example, Figure 2-5 illustrates a small COBOL procedure division that contains two modules. In this program segment there are 3 binary predicates and 2 connected components (modules) giving a cyclomatic number of 5.

```
PROCEDURE DIVISION.
PROCEDURE-1.
    IF A > B THEN
       ADD 10 TO B
    ELSE
       SUBTRACT 10 FROM B.
PROCEDURE-2.
    IF C EQUALS D THEN
       IF D > 20 THEN
          MOVE 20 TO C
       ELSE
          MOVE 30 TO C
    ELSE
       MOVE ZERO TO D.
```

**Figure 2-5.** A COBOL Program's Procedure Division

The results of McCabe's metric can be used in an operational environment: project members can be advised to limit their software modules by cyclomatic complexity in addition to physical size. Numerous studies [Wal79], [Hen81b], [War89], [Sch79], [Gib89], and [Hei94]

have shown that high measures of McCabe's cyclomatic complexity number correlate with high levels of errors in software. For example, Walsh [Wal79] looked at 276 procedures to see how cyclomatic complexity effected errors in the code. He found that modules with a cyclomatic number of 10 or higher experienced 21 percent more errors than those modules with a cyclomatic number less than 10. Approximately half of the procedures he analyzed had a cyclomatic number of 10 or greater.

These studies have shown that the more complex a module is, the more likely it will contain errors. Beyond a certain threshold of cyclomatic complexity, the chances are much greater that a module will contain errors. The particular threshold, or upper bound, of cyclomatic complexity that McCabe established is 10. McCabe's intention was to keep the complexity of the modules manageable and allow for testing all the independent paths. Limiting the cyclomatic complexity of software modules would increase the overall reliability of the software [Wal96]. This limit is waived in situations with single multiway decision (case) statements [Mcc76].

In studying testing and cyclomatic complexity, Wallace et al. cited that there are several good reasons to limit cyclomatic complexity. Modules that are highly complex are more likely to have errors, are more difficult to understand, are harder to test, and require more work to modify. Studies have shown that modules with high cyclomatic complexity correlated strongly with difficulty in understanding and modifying software [Cur79]. Purposely setting limits of cyclomatic complexity at every stage of the software development process helps avoid the difficulties associated with highly complex software. Several organizations have implemented cyclomatic complexity limits to their software. The exact number to use as a limit, however,

remains debatable. As stated earlier, McCabe originally proposed a module limit of 10, but limits as high as 15 have been successfully used as well [Wal96].

#### *2.3.2.2 Nesting Level*

Nesting enables programmers to avoid writing an "if" or loop statement with an inordinate amount of compound conditionals. This is possible because nested statements use the conditions of previous "if" or loop statements in the nested structure. Although this allows individual "if" or loop statements to be more manageable, an excessive amount of nesting can make it difficult for programmers to determine what conditions must be met before a statement is executed [Con86]. Nested statements may not be easy to simplify in complex programs that require satisfaction of many conditions before execution of an operation. However, if possible, one should try to reduce the level of nesting by using simple Boolean expressions [Ada92]. Also, case statements should be used instead of nested If-Statements when possible.

To determine the nesting level of any given module, each statement in a structure must be assigned a nesting level. This can be calculated by recursively carrying out the following procedure on a structure [Con86]:

1) The first executable statement has nesting level 1.

2) If statement *a* is at level *i* and statement *b* simply follows sequentially after statement *a*, the nesting level of statement *b* is also equal to *i*.

3) If a statement is a loop or conditional statement at level *i*, the next statement will be at level $i + 1$.

Figure 2-9 is an example of a COBOL structure with extensive nesting. The nesting level of each statement is listed to its right.

```
                                                    Level
IF WS-COUNT > 10                                      1
   IF WS-FLAG = "TRUE"                                2
      IF WS-PROFIT < 500                              3
         PERFORM 100-CLOSE-SHOP                       4
      ELSE                                            3
         IF WS-PROFIT > 5000                          4
            PERFORM 200-INVEST                        5
         ELSE                                         4
            PERFORM 300-KEEP-GOING.                   5
```

**Figure 2-6.** An Extensively Nested COBOL Structure

Although McCabe makes no distinction between the cyclomatic complexity of a nested If-Statement and a case statement, he implies that a case structure is much easier to understand than a nested If-Statement [McC76]. Shepperd [Shep88] also cites that because of this, a case statement shouldn't be considered as complex as a nested If-Statement when it comes to cyclomatic complexity. It has been suggested that a case statement contribute only one to a module's cyclomatic complexity, or perhaps a $\log_2(n)$ relationship, where $n$ is the number of cases.

#### *2.3.2.3 Information Flow Complexity Measure*

The Information Flow Complexity Measure introduced by Henry and Kafura measures the interprocedural complexity of a program [Hen81], which considers the complexities of procedures and modules within a system as well as the complexities of the interfaces between different components of the system. This measure looks to identify poor functional decomposition of procedures and modules, improper modularization, poorly designed data structures, and modifiability [Hen79]. The formula that computes the information flow measure of a module is illustrated in Equation 2-4.

$$Cp = length * (fan\text{-}in * fan\text{-}out)^2 \quad (2\text{-}3)$$

where:

*length* = module length.

*fan-in* = number of local flows into a module plus the number of data structures from which a module retrieves information.

*fan-out* = number of local flows from a module plus the number of data structures which the module updates.

The length of the module can be represented by LOC, number of statements, or function points, etc. It is not important how length is represented as long as it is always represented the same way when analyzing and comparing modules.

A local flow of information takes place from a module *A* to a module *B* if one or more of the following conditions take place:

1. If module *A* calls module *B*.
2. If module *B* calls module *A* and module *A* returns a value to module *B*, which module *B* uses.
3. If module *C* calls both modules *A* and *B* passing an output valued from module *A* to module *B*.

Figure 2-7 is an example of a COBOL procedure that calls two other procedures: 100-CHECK-IT and 200-UPDATE-FILE. Assume the procedure shown in Figure 2-7 is called by two different procedures: 025-GET-COUNT and 050-ADD-IT. 000-COUNT-IT retrieves information from one data structure: WS-COUNT-INC, and it updates two unique data structures: WS-COUNT-INC and WS-FLAG-CHECK. Figure 2-8 illustrates the calls to and from the procedure with a structure chart. Because COBOL uses global data structures and does

not pass variables when calling other procedures, only condition 1 of the above definition applies when calculating local flow.

```
000-COUNT-IT
   ADD 1 TO WS-COUNT-INC.
   IF WS-COUNT-INC > 5
      PERFORM 100-CHECK-IT
      MOVE "TRUE" TO WS-FLAG-CHECK
   ELSE
      PERFORM 200-UPDATE-FILE.
      MOVE "FALSE" TO WS-FLAG-CHECK.
```

**Figure 2-7.** An Example COBOL Procedure

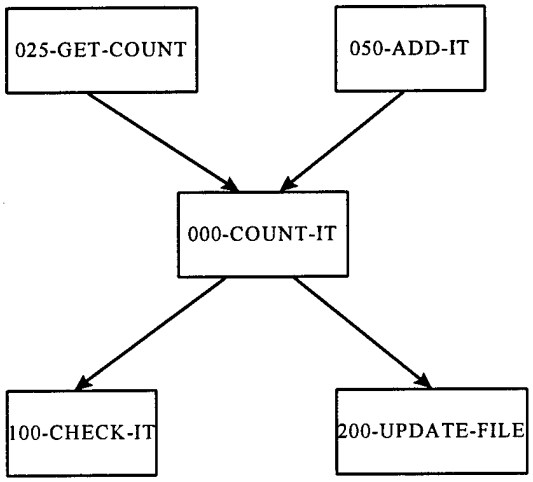



**Figure 2-8.** Structure Chart for Figure 2-7

The following illustrates how the Information Flow Complexity Measure is calculated for Figure 2-7:

| | |
|---|---|
| length = 6 | 6 statements |
| fan-in = 3 | 2 calls into the procedure |
| | 1 data structure it retrieves information from |
| fan-out = 4 | 2 calls from the procedure |
| | 2 data structures are updated |

Using equation 2-4:

$$Cp = 6 * (3 * 4)^2$$
$$Cp = 864$$

Henry and Kafura [Hen79] used the Information Flow Complexity Measure to determine if a correlation existed between the information flow measure and process measures. Specifically, they looked at the number of changes made on the system. Initial work found a strong correlation between modules with a high Information Flow Complexity Measure number and the number of changes made to each of those particular modules or procedures. The study further revealed a correlation between defects and their Information Flow measure [Hen79].

### *2.3.3 Product Measures Estimating Defects*

A study by Henry et al. [Hen79] and many studies using McCabe's Cyclomatic Complexity measure [Wal79], [Hen81b], [War89], [Sch79], and [Hei94] have found that the particular complexity product metrics studied correlated with defects found in the code. One of the hypotheses of this research is that a positive correlation would be found between defect data and program complexity. If it can be determined early in the development lifecycle which modules may end up being highly complex, these modules can be redesigned to try to reduce the complexity. Also, if it's not feasible to change the modules and we have empirical evidence that suggests a correlation between highly complex modules and defects, we can use this information

to predict extra maintenance efforts or schedule slippage that may occur due to the anticipated defects.

## *2.4 Conclusions*

Although product metrics have existed for over twenty years, evidence of their use by software organizations in the DoD is very scarce. The COBOL programming language is an old language and most of the systems developed with COBOL are also old. As stated earlier, very little work has been done with metrics on COBOL. This means very little empirical data is available that could help COBOL system managers use metrics to improve processes. The Air Force maintains millions of lines of COBOL code on hundreds of vital computer systems. These systems require recurring maintenance throughout their lifecycle. Over time, modules or procedures in the system can become more and more complex. Once this happens, a system can get caught in a vicious circle where programmers fix a problem, but increase the complexity. The increased complexity introduces yet more problems, and these problems must be fixed. This can continue until a new design is developed and released.

Using software product metrics to predict, and therefore reduce, future life-cycle problems would be a valuable asset to everyone that is affected by software. Moreover, by detecting problems at the root, managers and developers can produce higher quality software that will be more reliable throughout its lifecycle.

## *2.5 Summary*

This chapter provided detailed background information on several software product measures derived from source code. The following chapter describes the methodology used to

collect and use the product measures as well as how the data was analyzed once all the measures were collected.

# 3. *Methodology*

This chapter discusses the methodology used to collect and analyze the COBOL source code product measures and process measures. One goal of this research was to analyze a large COBOL legacy system that had at least 200K lines of code. Another requirement was that complete, previous versions of the source code had to be available so comparisons could be made between each release.

Source code measures were gathered from each of four consecutive major releases of the code. This was done so three major revisions could be observed. The source code measures gathered from a particular release were compared to the previous release's source code measures to observe any changes.

In the research, four consecutive major releases of the source code were analyzed from a COBOL system that contained over 400K LOC. This system, the Headquarters On-line System for Transportation (HOST), is currently maintained by the Air Mobility Command (AMC) / Computer Systems Squadron (CSS) at Scott AFB, Illinois. The system is made up of eight major subsystems. Five of these subsystems were analyzed. Each of these subsystems is made up of several COBOL programs. The five subsystems contained a total of 183 programs, which in turn consisted of over 5,500 modules.

The first step in this research was to decide which source code measures would be collected, then extract these measures from the source code using an automated tool, and finally deriving sample correlation coefficients and computing 95% confidence intervals for the correlation coefficients that showed a moderate or strong linear correlation.

### *3.1 Source Code Measures Selected*

Size and complexity measures were selected as the measures to be gathered. The size measure was chosen because it has proven to be a dominating factor for estimating software development effort [Fen97]. Several studies [Hen79], [Wal79], [Hen81b], [War89], [Sch79], and [Hei94] have found correlations between complexity measures and defects. This research investigated whether the size and complexity measures could be used to return similar results for the COBOL system being analyzed.

The raw product measures gathered and computed can be found in Appendices D and E. The following specific size measurements were gathered:

1. number of statements per module,
2. number of statements per program,
3. average number of statements per module,
4. number of modules per program, and
5. largest module per program.

The types of complexity measures gathered were McCabe's Cyclomatic Number, maximum nesting level, and Henry and Kafura's Information Flow. The following specific complexity measurements were gathered:

1. cyclomatic number, information flow, and maximum nesting level per module;
2. cyclomatic number, information flow, and maximum nesting level per program;
3. average cyclomatic number, information flow, and maximum nesting level per module; and
4. module with highest cyclomatic number, information flow, and maximum nesting level in program.

### *3.2 COBOL "EXIT" Procedures*

COBOL developers are routinely directed to use exit paragraphs or procedures at the end of each module so that the end of the module is clearly marked in the program. Exit procedures perform no processing and are used just to provide a clean termination of a procedure. An exit

procedure can only contain the exit statement; no other statements are allowed in an exit procedure.

The HOST software system contained exit procedures to terminate virtually every regular procedure. Because exit procedures contain only an exit statement and do no processing, these procedures were excluded from the measures collection and analysis. This reduced the amount of modules to be analyzed to approximately half.

### *3.3 Application Used to Derive Measures*

To automate the process of gathering product measures from a program's source code an application is needed that should have the following three capabilities:

1. A lexical analyzer created specifically for the program language being analyzed: The lexical analyzer parses the program's source code looking for predefined keywords or parts of grammar. Each time the lexical analyzer finds a keyword or grammar it creates a representation or "token" for that keyword or grammar.

2. The ability to create an Abstract Syntax Tree (AST): Once the tokens have been created, relationships between the tokens must be established. This is done by creating an AST. Figure 3-1 is a pseudo-code representation of an IF-THEN-ELSE statement. A primitive representation of an AST for the statement is illustrated in Figure 3-2.

```
If a > b then
  If b > c then
    a := 10;
  else
    a := 20
else
  a := 15
end if;
```

**Figure 3-1.** An If-Statement

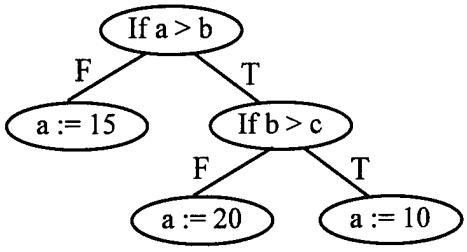



**Figure 3-2.** An If-Statement AST

3. A measurement tool: After the ASTs have been created, each tree must be analyzed to count the particular attributes of interest. A knowledge-based system must be used that understands the syntax of the specific language being analyzed as well as how the AST is being represented. This system must take into account all the programming structure rules associated with the language in order to gather the measurements correctly.

The automated tool used for this research was *Refine/COBOL*™. It provided the first two capabilities described above. To accomplish the third step, additional *Refine/COBOL*™ code had to be written.

*Refine/COBOL*™ is an interactive workbench that supports key tasks in working with COBOL source code [Rea95]. It was used to extract the product measures automatically from the COBOL source code. *Refine/COBOL*™ uses the *Refine*™ programming language. This programming language supports set theory, logic, transformation rules, pattern matching, among others [Rea90]. *Refine/COBOL*™ uses these features to analyze the logic, data components and relationships in a COBOL program and creates an Abstract Syntax Tree (AST). Once the AST is built for a program, *Refine/COBOL*™ code can be written to extract specific data from the AST.

Figure 3-3 is an example COBOL If-Statement. Figure 3-4 is an AST subtree that represents the specific COBOL If-Statement in Figure 3-3. *Refine/COBOL*™ breaks the

statement down, in this case the If-Statement, into its smallest parts while maintaining the relationships between each part of the statement.

```
IF CUSTOMER-STATUS OF ORDER-OUT > 1 THEN
   SUBTRACT 1 FROM CUSTOMER-STATUS OF ORDER-OUT.
```

**Figure 3-3.** A COBOL If-Statement

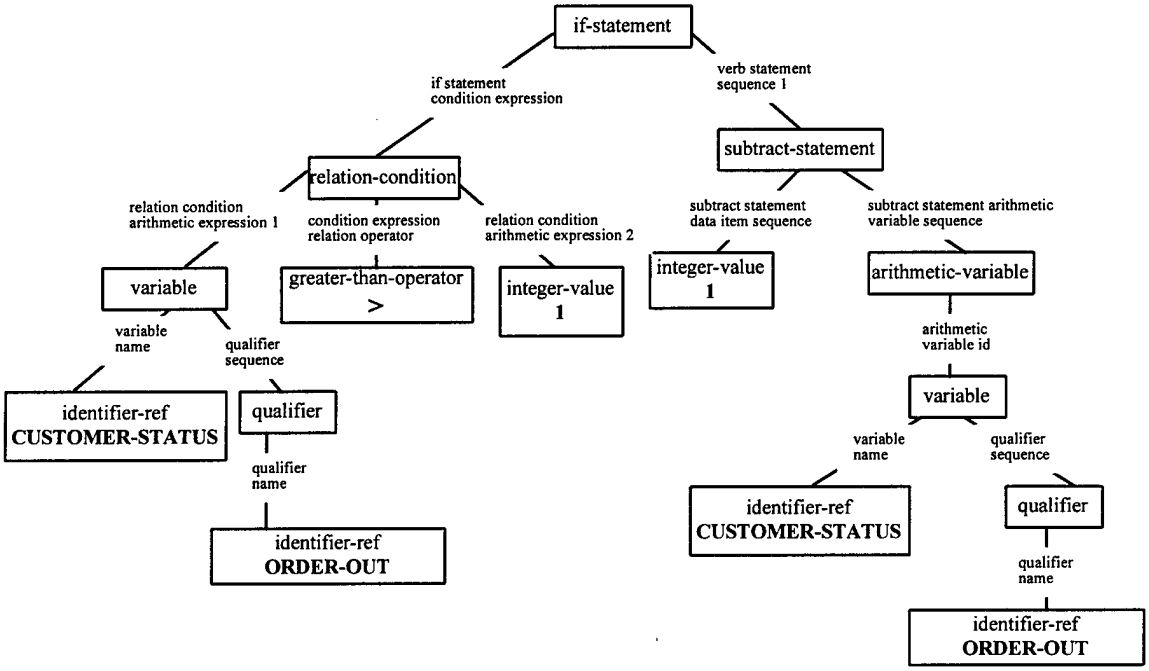



**Figure 3-4.** An AST of an If-Statement

Not all statements of the same type have identical subtrees; the AST that is built represents the specific code being analyzed. For example, two If-Statements may have completely different ASTs depending on the complexity of each statement. The AST in Figure 3-6 represents the slightly more complex If-Else-Statement shown in Figure 3-5. Compared with Figure 3-4, many more parameters are involved.

```
IF GREEN-WIDGETS OF ORDER-IN = ZERO OR
   RED-WIDGETS OF ORDER-IN = ZERO THEN
      MULTIPLY .93 BY WIDGET-PRICE OF ORDER-OUT
ELSE
      MOVE ZERO TO RED-WIDGETS OF ORDER-IN.
```

**Figure 3-5.** A More Complex COBOL If-Statement

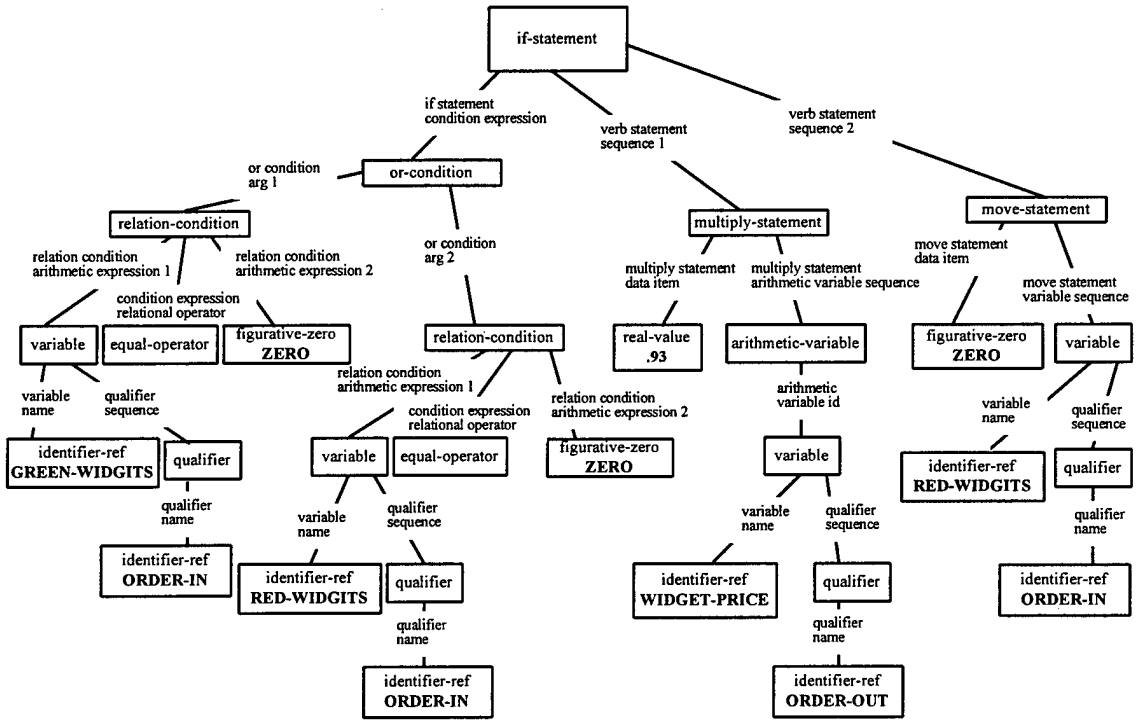



**Figure 3-6.** An AST of a More Complex If-Else-Statement

Because of these differences, the *Refine/COBOL*™ code that was developed to gather measures from the ASTs was written to handle the different configurations a COBOL statement's AST can entail. All of these possible configurations are defined in the

*Refine/COBOL™ User's Guide* [Rea95]. The *Refine/COBOL™* code that was written to compute and gather source code measures can be found in Appendix A.

### *3.3.1 Computing Size Measures*

As stated in Chapter 2, there are a variety of ways to measure the size of a module. The most popular means of measuring size is lines of code (LOC). Because the automated tool being used did not have the capability of extracting LOC from the source code, the number of statements per module was used instead. This also seemed to be a better measure for observing any changes in the code because statements can sometimes take up several lines of code. For example, if a statement was added, deleted, or changed it would not show bias to larger statements that may take up several LOC or a smaller statement that may take up only one LOC.

### *3.3.2 Computing McCabe's Cyclomatic Number*

McCabe's Cyclomatic Number was derived by writing code that analyzed each conditional or loop statement in a module and deriving the overall cyclomatic number for the module by using the formula described in Chapter 2 of this document.

### *3.3.3 Computing Henry and Kafura's Information Flow Measure*

The Information Flow Measure was derived by writing code that analyzed every statement in the module to determine whether or not variables were being used or updated in the statement. COBOL uses only global variables in a program and does not pass variables back and forth between procedures. Because of this, the scope of determining the local flow of information from or to a procedure was narrowed to using only the first condition of what constitutes a local flow described in Chapter 2.

### *3.3.4 Computing Maximum Nesting Level*

The maximum nesting level of a module was determined by writing code that analyzed each conditional and looping statement in the module and determining which of these statements had the highest nestling level as defined in Chapter 2.

## *3.4 Gathering the Process Measures and Source Code Measures*

AMC/CSS provided four complete, consecutive versions of the five subsystems for analysis. The earliest version of the code received was established as the baseline version. The three subsequent versions were considered major releases. All measures were gathered and organized separately by release.

### *3.4.1 Gathering Process Measures by Release*

All changes that are made during each major release are dictated by either a Baseline Change Request (BCR) or a System Problem Report (SPR). BCRs are initiated when users need to add to, or change, the functionality of the system, or when users find problems with the system. SPRs are initiated internally in the development organization when a problem is found in the system. For the HOST system, BCRs and SPRs are organized by subsystems. In other words, a BCR or SPR will be associated with only one subsystem. If a change is needed that will modify more than one subsystem, a separate BCR or SPR will be drawn up for each subsystem.

All the BCRs and SPRs associated with the three major releases and which were analyzed in this research were provided by AMC/CSS. In addition, an Impact Analysis Worksheet (IAW) accompanied each BCR and SPR. An IAW estimates how many programs will need to be added, if any, and which programs will need to be changed. It also gives estimates on the number of LOCs that will be added, changed, or deleted, and the estimated

number of manhours that will be required to make the additions or changes. AMC/CSS takes this a step further after the additions or changes have been made by providing the actual programs added or changed including the LOCs added, changed, or deleted as well as the total number of manhours it took to accomplish the task for each program.

### *3.4.2 Gathering Source Code Measures by Release*

All code from the baseline version was analyzed and source code measures were gathered and entered into a spreadsheet. The IAWs provided a detailed list of all the programs that were changed or added in each major release. Therefore, it was only necessary to collect and analyze the source code measures from the programs that actually were changed in each major release. Once all the source code measures for each of the major release were gathered, they were entered in a spreadsheet along with the baseline's source code measures. Differences in the product measures could then be easily observed between the releases.

#### *3.4.2.1 Additional Source Code Measures*

Although the IAWs provided the number of LOCs that were changed, added, or deleted in a program, the data could not be used for the purpose of this research. This is because this research was interested only in the *Procedure Division* of the COBOL programs. LOC modifications listed in the IAW include all divisions of the COBOL program. Hence, each program that was changed was examined with a text editor to see which actual changes were made only in the *Procedure Division* of each program. Once these were gathered, they were also entered into a spreadsheet along with the source code measures.

### *3.5 Comparing Measures Gathered*

The *Statistix*® software package was used to compute the correlations between sets of data. Scatter plot were constructed and sample correlation coefficients were computed for the compared measures. This was done to see if any positive correlations existed between process measures such as defects or manhours with product measures derived from the source code such as size or complexity. A high positive correlation between measures means that a high number in a one measure will result in a relatively high number in the other measure and vice versa.

When trying to draw relationships between data, constructing a scatter plot should be the first step. Visually inspecting the data with a scatter plot can often reveal features that could be overlooked if only tabular data was examined [Dev95], [Ana96]. The scatter plot should give insight on whether there is a correlation between the data and whether the correlation is positive or negative.

After constructing scatter plots for all the correlations, the *Pearson* Correlation was used to compute the sample correlation coefficient between the sets of data. The *Pearson* correlation returns a sample correlation coefficient $r$ which measures the degree of linear relationship between the set of data [Ana96]. Devore [Dev95] states that the strength of the correlation between a set of data is said to be

- strong if $.8 \leq |r| \leq 1$,
- moderate if $.5 < |r| < .8$, and
- weak if $0 \leq |r| \leq .5$.

### *3.6 Computing Confidence Intervals*

Confidence intervals were computed for compared measures that had a moderate or strong sample correlation coefficient. Normally, confidence intervals can only be computed on data that have a normal population distribution and for which the standard deviation is know for

the population. However, these requirements can be waived by using the Central Limit Theorem when a large sample of data is being measured. Generally, a sample of data is considered large if the population is > 30 [Dev95].

For the purpose of this research, the sample correlation coefficient, *r*, and the sample data size is used to determine the 95% population confidence interval for each compared measure that had a moderate or strong sample correlation coefficient. The method used for computing the confidence interval was taken from Devore's *Probability and Statistics* text [Dev95].

### *3.7 Problems with Defect Data*

This research initially focused on defect data as the primary process measure for which product measures would be correlated against. However, because the defect data was recorded at a level that did not map to specific programs or code, the data could not be used to draw any correlations with product measures. The following explains this further.

AMC/CSS handles each BCR and SPR as a complete software development cycle including: analysis, preliminary design, detail design, coding, unit testing, and formal testing. They record defect data by BCR or SPR and prioritize the defect according to MIL-STD-498 [Mil94] as illustrated in Table 3-1. The defects are categorized according to phase of the development cycle in which they were detected. Appendix C contains the defect data provided by AMC/CSS. It displays defects by release and BCR or SPR. It also shows which stage of the development cycle the defect was detected.

Because the data provided was at such a high level, it could not be determined which program or module was the cause of the defect. Therefore, this research was unable to use the defect data to draw any correlations with other measures.

**Table 3-1.** Defect Classifications According to MIL-STD-498 [Mil94]

| Priority | Applies if a problem could: |
|---|---|
| 1 | a. Prevent the accomplishment of an operational or mission essential capability<br>b. Jeopardize safety, security, or other requirement designated "critical" |
| 2 | a. Adversely affect the accomplishment of an operational or mission essential capability and no work-around solution is known<br>b. Adversely affect technical, cost, or schedule risks to the project or to lifecycle support of the system, and no work-around solution is known |
| 3 | a. Adversely affect the accomplishment of an operational or mission essential capability but a work-around solution is known<br>b. Adversely affect technical, cost, or schedule risks to the project or to lifecycle support of the system, but a work-around solution is known |
| 4 | a. Result in user/operator inconvenience or annoyance but does not affect a required operational or mission essential capability<br>b. Result in inconvenience or annoyance for development or support personnel, but does not prevent the accomplishment of those responsibilities |
| 5 | a. Any other effect. |

### *3.8 Comparing Manhours with Product measures*

The research also focused on the manhours measures reported on the IAWs. Manhours were mapped to each program that was changed or created. The manhour data did not map to lower levels such as the modules or statements of the program. Ergo, each correlation was computed by using the program totals for each of the product measures. The specific manhour correlations derived as well as the results and discussions of the manhour correlations are presented in Chapter 4.

### *3.9 Comparing Product Measures with Product measures*

This research also investigated correlations between product measures. It investigated whether the product measures derived from the source code correlated with one another. The

product measures were derived from the source code at the module level and compared at the module level. Two different types of product measure comparisons were investigated.

The first type, derived from the baseline version of code, reflects the status of the code before any changes were made in subsequent versions. This comparison looked at product measures from every module in the systems and derived sample correlation coefficients between the product measures. For example, it investigated whether the size of a module positively correlated with the cyclomatic complexity of the module.

The second type, derived from the changed or added modules from the three revisions, investigated whether changes in one source code measure effected changes to other source code measures. This comparison looked at changes for each of the product measures for every module in the systems that experienced a change. It took the differences from each product measure between revisions and derived sample correlation coefficients for the differences. For example, it investigated if a change in the size of a module positively correlated with a change in the cyclomatic complexity of the module.

The specific product correlations derived as well as the results and discussions of the product measure correlations are presented in Chapter 4.

### *3.10 Summary*

This chapter has discussed the methodology used to collect the software product and process measures. It outlined the types of measures that were examined for correlations as well as how the degree of correlations were computed. It also explained how confidence intervals were computed for the compared data that had moderate or strong sample correlation coefficients. The specific measures as well as the results and discussions of this methodology are presented in Chapter 4 of this paper.

# 4. Data Analysis and Results

This chapter reports and discusses the findings of the methodology chapter of this thesis. It displays the scatter plots and the degree of linear correlation for each of the compared measures. 95% confidence intervals were computed for those comparisons that had a moderate or strong sample correlation coefficient. The measures compared, the sample correlation coefficients, and the confidence intervals will be summarized in a table for each category of measures. In each table, $r$ represents the sample correlation coefficient; $CI_L$ and $CI_U$ represent the lower and upper bound confidence intervals, respectively.

This chapter will then elaborate on the findings and give some insight on what the findings may indicate. These findings may then be used to determine whether AMC/CSS should consider modifying certain processes in their measurement procedures.

## 4.1 Using Defect Data for Prediction

From the onset, one of the hypotheses of this research was that a positive correlation could be found between defect data and program complexity. A study by Henry et al. [Hen79] and several studies using McCabe's Cyclomatic Complexity measure [Wal79], [Hen81b], [War89], [Sch79], and [Hei94] discovered correlations between the particular complexity metric being used and defects found in the specific systems analyzed. If this research could have shown a positive correlation between defect data and complexity measures, it could have provided AMC/CSS with information that could be used for predicting later lifecycle workloads. For instance, if it could be shown that higher complex modules resulted in higher defects in the code, this information could possibly be used to predict a high number of errors for highly complex

modules. In turn, as Jones states [Jon91], this could be used to predict maintenance efforts, cost overruns, or schedule slips that occur later in the software's lifecycle for highly complex modules. However, the defect data collected by AMC/CSS did not map to the specific code or program that caused the defect. Because the defects did not map to code, this research was unable to investigate relationships between the defect data and the complexity of the code.

### *4.2 Manhour Data Comparisons*

The following definitions apply for the product measures used to compute the degree of correlation with the manhours measure:

- *Affected Statements*: The total number of statements that were either added, changed, or deleted in the program. For example if a module had two statements added, two statements deleted, and two statements changed, the total "affected statements" is six.
- *Pre-Cyclomatic Complexity*: The total cyclomatic number of the program before the change.
- *Pre-Information Flow*: The total information flow of the program before the change.
- *Pre-Number of Statements*: The total number of statements of the program before the change.
- *Post-Cyclomatic Complexity*: The total cyclomatic number of the program after the change.
- *Post-Information Flow*: The total information flow of the program after the change.
- *Post-Number of Statements*: The total number of statements of the program after the change.
- *Change in Cyclomatic Complexity*: The net change in cyclomatic complexity for a module (can be negative).
- *Change in Information Flow*: The net change in information flow for a module (can be negative).
- *Change in Program Size*: The net change in the number of statements for a module (can be negative).

Scatter plots were constructed and sample correlation coefficients were computed for the following four categories:

1. Manhours vs. Affected Statements
2. Manhours vs. Post[1]-Cyclomatic Complexity
3. Manhours vs. Post[1]-Information Flow
4. Manhours vs. Post[1]-Number of Statements

These measurements were taken for 39 programs that were either changed or added during the three version releases of the software observed. The raw product measurement data used for computing the degree of correlations can be found in Appendix G.

Table 4-1 illustrates the sample correlation coefficients for the manhour and product measures outlined above. It also shows the 95% confidence intervals for the sample correlation coefficient that showed a strong linear correlation between the measures. Figure 4-1 is a scatter plot that illustrates the linear relationship between manhours and affected statements. The scatter plots for the remaining correlations can be found in Appendix H.

**Table 4-1.** Initial Manhour Sample Correlation Coefficients & CIs

| *r* | $CI_L$ | $CI_U$ | Source Code Measure |
|---|---|---|---|
| .94 | .88 | .97 | Affected Statements |
| .39 | - | - | Post-Cyclomatic Complexity |
| -.03 | - | - | Post-Information Flow |
| .48 | - | - | Post-Number of Statements |

[1] Only "Post" measures were observed because some code (new code) did not have "Pre" measures.

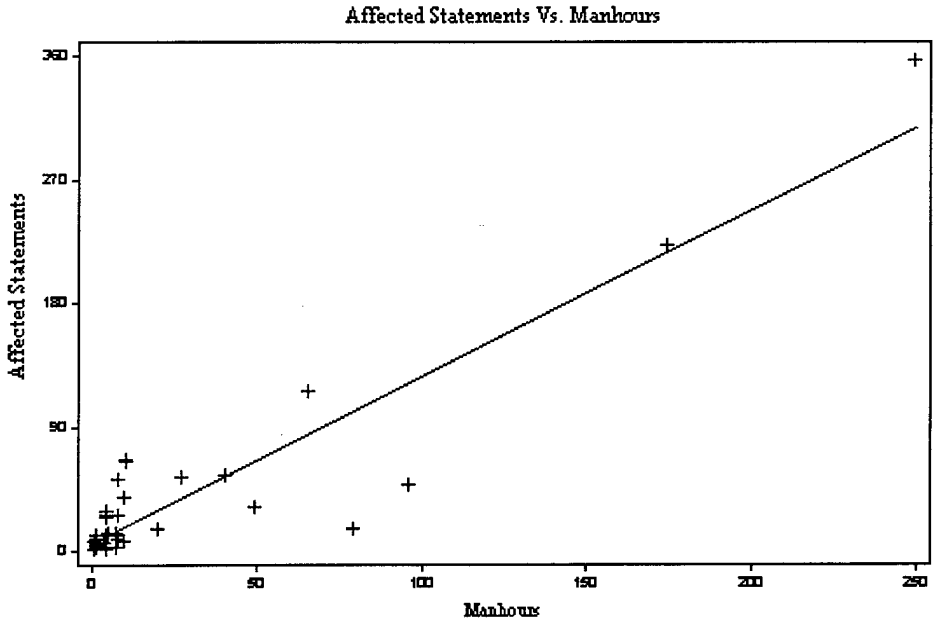



**Figure 4-1.** Scatter Plot for Manhours Vs. Affected Statements

### *4.2.1 Comparing Manhours with Affected Statements*

If empirical data can be kept for ongoing development projects for the number of manhours it actually takes to produce code, this data could be used to help managers estimate future projects of similar types more precisely. With this in mind, this research attempted to see if the manhour data being collected by AMC/CSS along with the affected statements data derived from the source code could be used for estimating effort in future projects. If a high, positive correlation existed between manhours and the number of statements affected during a change, these findings could be used as empirical data to provide AMC/CSS with information that may help them estimate the effort that will be needed on other projects of the same nature.

The findings show a very high, positive correlation with the number of statements affected and the number of manhours applied. It also showed a tight confidence interval suggesting that the correlation coefficient for the population would be strong. AMC/CSS estimates the number of affected statements a project will take. However, sometimes these estimates were somewhat inaccurate. This is to be expected because if is difficult to know exactly how many LOCs it will take to complete a change. However, if more accurate estimates of affected statements could be made, and knowing that manhours correlate highly with affected statements, this information could be used to estimate the amount of time that will be needed to complete a project.

### *4.2.2 Comparing Manhours with Complexity and Size Measures*

As Table 4-1 illustrates, only weak sample correlation coefficients were found with the manhour data against cyclomatic complexity, information flow, and number of statements. This was particularly disappointing for the complexity measures because this research had hypothesized that complexity would affect the time it took to make changes to the code. If it had turned out that the manhours were highly correlated with the complexity of the code it could serve a two-fold purpose:

1. It would give managers and developers estimating ability before code is even written. If it can be derived from the specifications how complex the code will be by determining the cyclomatic number or information flow, the information could be used to help estimate the effort needed, and

2. It could give managers and developers advanced warning of code that may take longer than expected to write. This could enable the developers to restructure the specifications in an attempt to reduce the complexity of the code.

This research looked at size vs. manhours to see if perhaps the size the module had an effect on the manhours. It investigated whether larger modules were more complex, therefore making them more difficult to change or extend. This comparison also turned up a weak linear correlation. Consequently, for the particular organization and application researched, no correlation was found between the size of a module and the manhours needed to modify it.

It is difficult to say why no correlations existed with this comparison. Brooks [Bro95] cites many factors that may influence the manhour tracking such as computer downtime, meetings, informal conversations, paperwork, personal time, and sickness, etc. If these factors, or others, were included in the identification of the manhours that went into a project, it would be very difficult to draw any correlations.

### *4.2.3 Further Analysis of Manhour Data*

Although not originally planned, further analysis was conducted on the manhours data by segregating the data from the revisions into two categories: new code and changed code. In other words, this research looked at the same types of correlations between manhours and the product measures investigated above; however, now the correlations were done on two different categories of the manhours data. There were 10 new programs added and 29 programs changed over the period of the three releases for a total of 39 programs. See Appendix G for the raw product measures.

## *4.3 Comparing Manhours with Only Changed Code*

There were 29 programs that were changed by direction of a BCR or SPR over the three revisions of the system observed in this research. Although the sample size was only 29, it is

considered large enough to use the Central Limit Theorem[2] and consider to have a normal distribution.

Scatter plots were constructed and sample correlation coefficients were computed for the following 7 categories for the new code:

1. Manhours vs. Affected Statements
2. Manhours vs. Pre[3]-Cyclomatic Complexity
3. Manhours vs. Pre[3]-Information Flow
4. Manhours vs. Pre[3]-Number of Statements
5. Manhours vs. Post-Cyclomatic Complexity
6. Manhours vs. Post-Information Flow
7. Manhours vs. Post-Number of Statements

These sample correlation coefficients accounted only for the code changed during the three version releases of the software observed. This segregation of the code was done to see if any additional information could be derived.

Because this dealt with strictly changed code, the analysis looked for correlations with the source code measures before and after the change. The raw product measures for the changed code can be found in Appendix G.

Table 4-2 illustrates the sample correlation coefficients for manhour and product measure data described in Chapter 3 of this paper. It also shows the 95% confidence intervals for the sample correlation coefficients that showed a moderate linear correlation between the measures. Figure 4-2 is the scatter plot illustrating the correlations between manhours and affected statements. The scatter plots for the remaining correlations can be found in Appendix H. The following sections will elaborate further.

---

[2] Generally, the Central Limit Theorem considers a data set large enough to represent a normal distribution if the data set is $> 30$.

[3] "Pre" measures were now correlated because new code is not being used in this category

**Table 4-2.** Sample Correlation Coefficients & CIs for Only Changed Code

| *r* | $CI_L$ | $CI_U$ | Measure |
|---|---|---|---|
| .52 | .19 | .75 | Affected Statements |
| .12 | - | - | Post-Cyclomatic Complexity |
| -.05 | - | - | Post-Information Flow |
| .10 | - | - | Post-Number of Statements |
| .08 | - | - | Pre-Cyclomatic Complexity |
| -.08 | - | - | Pre-Information Flow |
| .05 | - | - | Pre-Number of Statements |

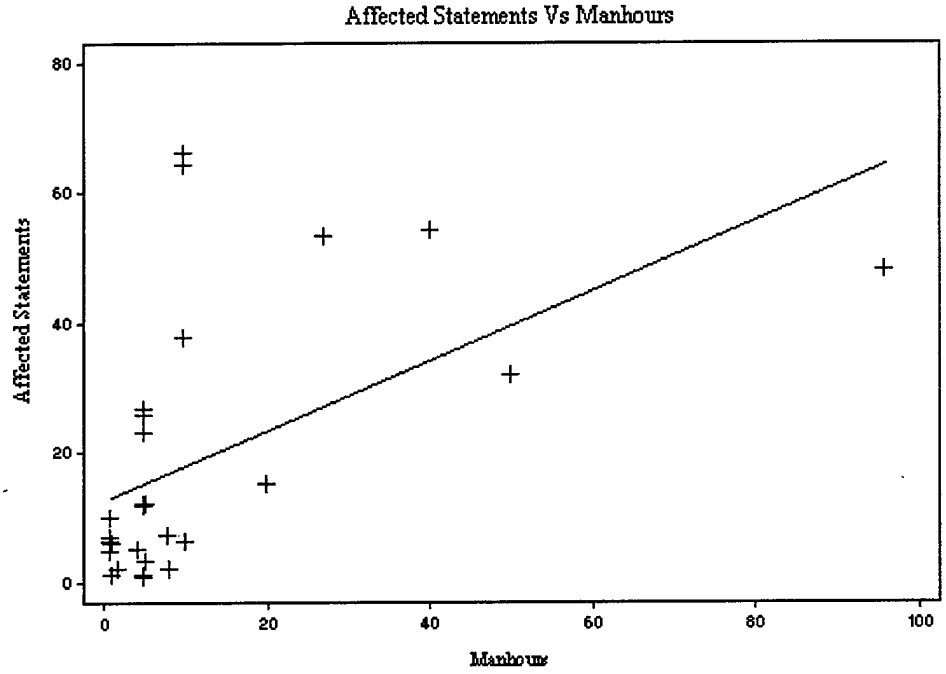



**Figure 4-2.** Manhours vs. Affected Statements (Changed Code)

### *4.3.1 Comparing Manhours with Affected Statements (changed code)*

The sample correlation coefficient between manhours and affected statements dropped significantly for changed code versus new code. It now showed only a moderate, bordering on weak, linear correlation. This suggested that the new code had a significant influence on the

high linear correlation of the two sets of data when the changed code and new code were correlated together in the previous section.

The 95% confidence interval also widened significantly. This reveals that it is unlikely that a moderate linear relationship exists for the correlation coefficient from the population of only changed code.

The fact that the new code had such a significant influence on the initial manhour comparisons could suggest that it is much easier to predict the time it takes to develop code from scratch versus changing existing code. This perhaps could be attributed to many factors which make it more difficult to make changes in existing code such as the complexity of the existing code, the size of the existing modules, determining how to extend the existing code, determining how modules are dependent upon one another, and cross-team communications, etc [Jon91].

### *4.3.2 Comparing Manhours with Complexity and Size (changed code)*

As Table 4-2 illustrates, only weak sample correlation coefficients were found with the manhour data against cyclomatic complexity, information flow, and number of statements for both before and after the change. These sample correlation coefficients were significantly weaker than when the sample correlation coefficients were computed with both the changed code and new code. This, once again, would suggest that the new code had a significant influence on the slightly higher (although still weak), positive correlation of the two sets of data when the changed code and new code were correlated together in the previous section. The reason for this is a topic for future research.

### *4.4 Comparing Manhours with Only New Code*

There were 10 programs added by direction of a BCR or SPR over the three revisions of the system observed in this research. Because the sample data set was so small, a normal distribution representation could not be assumed. This research computed the sample correlation coefficient and confidence intervals for these measures with a caveat that the data is not significantly large enough to conclude the sample data is a good representation of the population.

Because this category deals with only new code, no "changed" or "pre" data exists, hence it was only necessary to look at the created code. Scatter plots were constructed and sample correlation coefficients were computed for the following 3 categories for the new code:

1. Manhours vs. Number Statements
2. Manhours vs. Cyclomatic Complexity
3. Manhours vs. Information Flow

These comparisons account for only the new programs added during the three version releases of the software observed. Because this dealt with strictly new code, the analysis looked for correlations with the source code measures after the code was written. The raw product measures for the new code can be found in Appendix G.

Table 4-3 illustrates the sample correlation coefficients for manhour and product measure computed. It also shows the 95% confidence intervals for the sample correlation coefficients computed between the measures. Figure 4-3 through 4-5 are the scatter plots illustrating the linear relationship between these measures.

**Table 4-3.** Sample Correlation Coefficients and CIs for Only New Code

| $r$ | $CI_L$ | $CI_U$ | Measure |
|---|---|---|---|
| .99 | .96 | 1 | Number Statements |
| .98 | .91 | .99 | Cyclomatic Complexity |
| .83 | .42 | .95 | Information Flow |

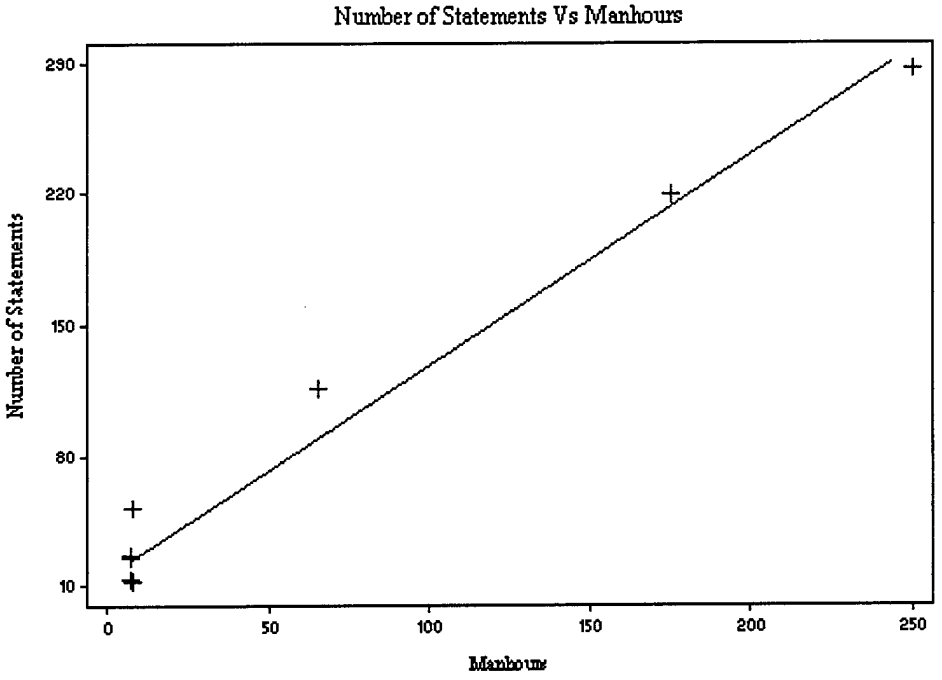



**Figure 4-3.** Manhours vs. Number of Statements (New Code)

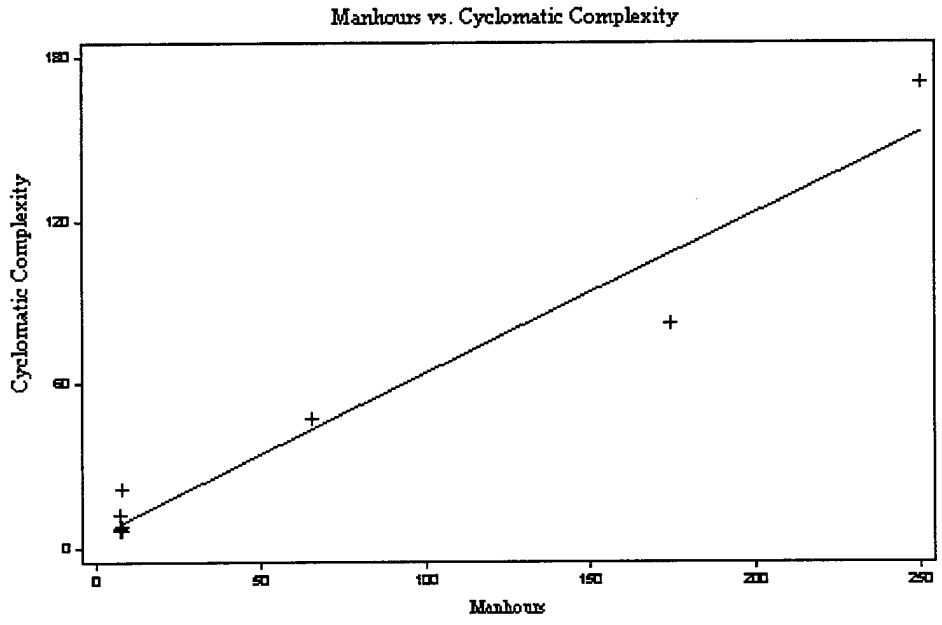



**Figure 4-4.** Manhours vs. Cyclomatic Complexity (New Code)

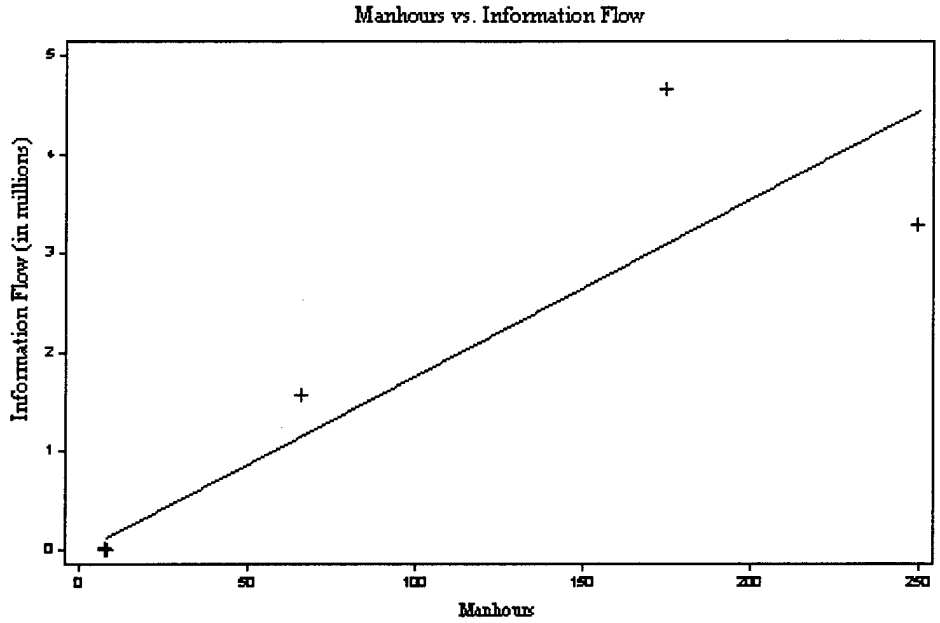



**Figure 4-5.** Manhours Vs. Information Flow (New Code)

The correlations with manhours vs. the number of statements, cyclomatic complexity, and information flow were very strong for the new code. However, to make inferences about data, the set of data must be large enough to represent the actual distribution of the data. The Central Limit Theorem [Dev95] states that the sample size should be at least 30 to give a distribution that reflects closely to the actual distribution. This sample size consisted of only of 10 programs. Therefore, we cannot make assumptions that these 10 programs are a statistically good representation of what to expect with all new code. However, from viewing the scatter plots in Figures 4-3 through 4-5, the data suggest that a strong trend exists between the compared measures, especially with the manhours versus the number of statements comparison. If the results were the same on a larger set of data (> 30) and the size of the code to be developed was estimated correctly, it could be inferred that AMC/CSS could accurately predict the time it takes

to develop new code; assuming these types of measures are routinely tracked within the organization.

### *4.5 Comparing Source Code Measures from the Baseline Code*

Source code measures were correlated for each module from the baseline version of code. The raw product measures can be found in Appendix B. Scatter plots were constructed and sample correlation coefficients were computed for the following 4 categories:

1. Cyclomatic Complexity with Number of Statements
2. Information Flow with Number of Statements
3. Cyclomatic Complexity with Information Flow
4. Cyclomatic Complexity with Nesting Level

Table 4-4 illustrates the product measure sample correlation coefficients found. It also shows the 95% confidence intervals for the sample correlation coefficients that showed a moderate linear correlation between the measures. Figure 4-4 is a scatter plot illustrating the linear relationship between cyclomatic complexity of a module and the module's number of statements. The scatter plots for the remaining correlations can be found in Appendix H. Excluding COBOL exit paragraphs, there were over 2700 modules used for the sample set. The raw product measures from the baseline can be found in Appendix B.

**Table 4-4.** Sample correlation coefficients from Baseline Source Code

| $r$ | $CI_L$ | $CI_U$ | Measure |
|---|---|---|---|
| .64 | .62 | .66 | Cyclomatic Complexity with Number Statements |
| .56 | .53 | .59 | Information Flow with Number Statements |
| .35 | - | - | Cyclomatic Complexity with Information Flow |
| .40 | - | - | Cyclomatic Complexity with Maximum Nesting Level |

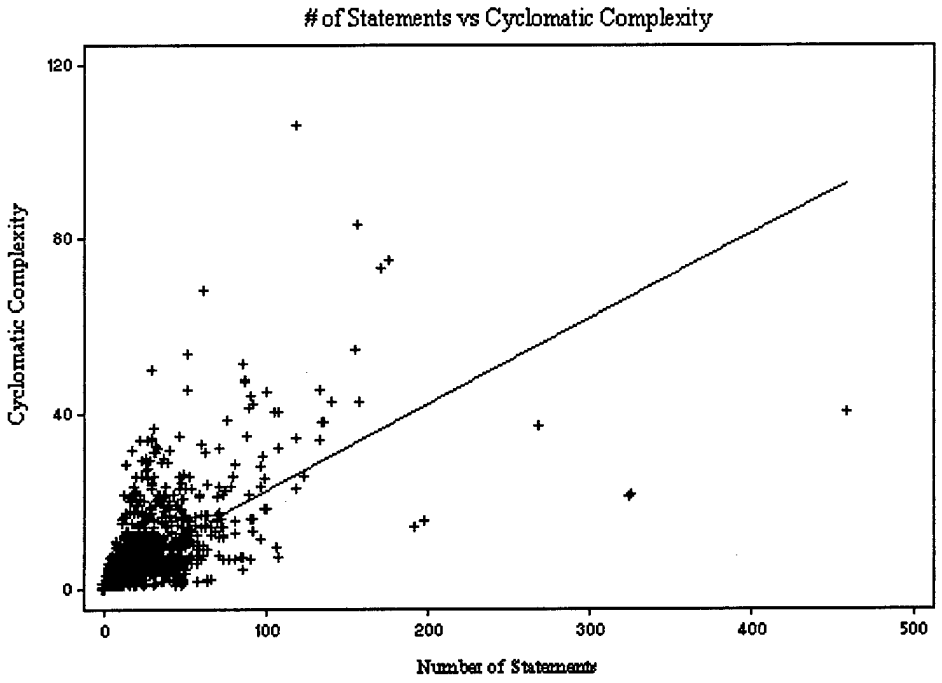



**Figure 4-6.** Product Measures of Size vs. Cyclomatic Complexity

### *4.5.1 Comparing Cyclomatic Complexity with Number of Statements*

Although McCabe [McC76] suggested that size and cyclomatic complexity measures are independent, this analysis showed a moderate linear correlation between size and cyclomatic complexity. Also, because the sample set was so large, the confidence intervals were very tight for this comparison. This suggests that the sample correlation coefficient is a good reflection of the population's correlation coefficient. This however, does not mean that the two measures are necessarily dependent upon one another. There were several modules throughout the code that had cyclomatic numbers of only 1, but which were relatively large.

### *4.5.2 Comparing Information Flow with Number of Statements*

The information flow measure had a moderate linear correlation with the number of statements. This would seem obvious because the size of the module (number of statements) is

used to compute the information flow measure. However, many other factors are involved with the information flow measure and it should not be proven that the measures compared are dependent upon one another. There were several modules of code that had an information flow of zero, but had up to 50 statements. This is because no variables were used or updated in the modules, resulting in the size not being a factor (See Equation 2-4 in Chapter 2).

### *4.5.3 Comparing Information Flow with Cyclomatic Complexity*

This correlation was an attempt to parallel Moe's work [Moe91] in which she attempted to use complexity metrics to measure the same piece of code to see if each measure returned the relatively same correlation strength. If, over time, these measurements returned relatively the same results on all source code, it would be redundant to use both product metrics in a measurement process. As shown in Table 4-4, this linear correlation was weak, concluding that, at least for the source code studied in this research, the information flow measure and cyclomatic complexity measure will not necessarily return the same degree of complexity on the same piece of code.

### *4.5.4 Comparing Cyclomatic Complexity with Maximum Nesting Level*

Nesting level is independent of cyclomatic complexity, but cyclomatic complexity is not independent of nesting level. This is because cyclomatic complexity can increase in a structure without increasing nesting level. This could be due to compound predicates, or the addition of smaller sequential decision statements to the structure. However, any time nesting level increases, cyclomatic complexity increases [Con86].

This correlation was an attempt to see if cyclomatic complexity correlated with the maximum nesting level of the module. This correlation turned out to be somewhat weak. This is

not too surprising as the correlation was using the entire modules cyclomatic complexity and correlating it with the one statement in the module that had the highest nesting level. To get a better assessment of this type of correlation, modules would have to be analyzed at the statement level and measure the degree of linear correlation between each statement's cyclomatic complexity and its nesting level. This comparison was not made because it is beyond the scope of this research.

### *4.6 Comparing Source Code Measures from the Changed Modules*

The following product measures were compared and sample correlation coefficients were derived for each module from the changed or added versions of code. These correlations were computed because, while analyzing the data, it appeared that when statements were added, deleted, or modified in a module, cyclomatic complexity changed as well. Scatter plots were constructed and sample correlation coefficients were computed for the following categories:

1. Change in cyclomatic complexity with change in number of statements
2. The absolute value of the change in cyclomatic complexity with affected number of statements
3. The absolute value of the change in the number of statements with affected number of statements

Table 4-5 illustrates the product measure sample correlation coefficients found. It also shows the 95% confidence intervals for the sample correlation coefficient for each compared measure. Figure 4-5 is a scatter plot illustrating the linear correlation between the change in cyclomatic complexity of a module with the change in the module's number of statements. The scatter plots for the remaining correlations can be found in Appendix H. See Appendix F for the

raw product measures used for these correlations. The sample size consisted of 210 compared modules.

**Table 4-5.** Sample correlation coefficients from Changed Modules Source Code

| *r* | $CI_L$ | $CI_U$ | Measure |
|---|---|---|---|
| .81 | .76 | .85 | Change in cyclomatic complexity with change in number of statements |
| .98 | .97 | .98 | Absolute value of change in the number of statements with affected number of statements |
| .80 | .74 | .84 | Absolute value of change in cyclomatic complexity with affected number of statements |

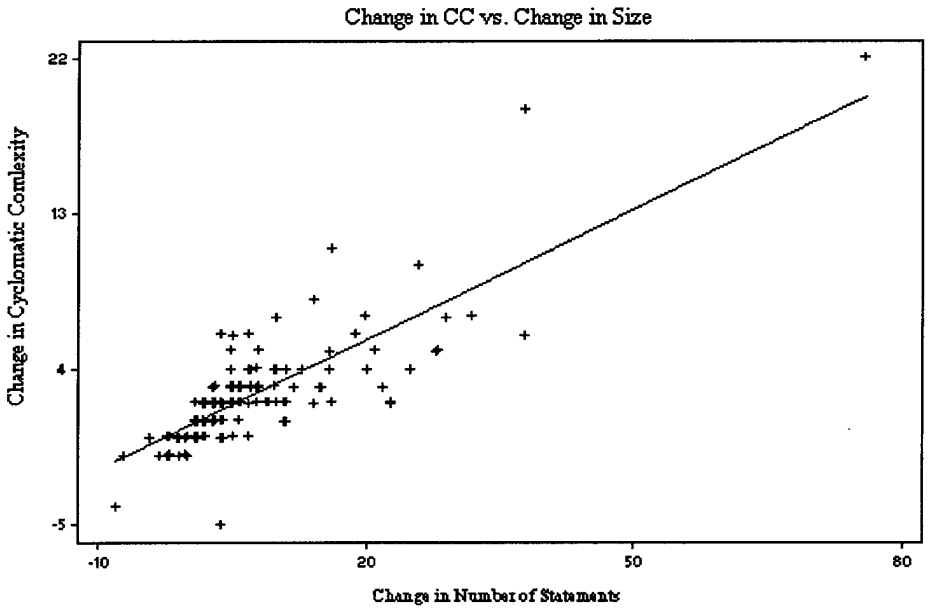



**Figure 4-7.** Change in Size with Change in Cyclomatic Complexity

### *4.6.1 Comparing Change in Cyclomatic Complexity with Change in Size*

During the analysis of the changes in code, it was observed that cyclomatic complexity changed often when code was modified. This correlation was computed to determine if there was a linear relation between changes in size with changes in cyclomatic complexity. This information could be used for anticipating an increase in cyclomatic complexity when new or changed code is introduced.

The sample correlation coefficient suggests a high, positive correlation between the data. This is a surprising finding because one would not assume that cyclomatic complexity would correlate so strongly with changes in size. This could perhaps be a unique circumstance where the changes made in these three revisions predominantly required the addition or deletion of decision statements. The changes or additions being made during these revisions could reflect a certain class of fixes that address exceptions (i.e. if <condition>) that were either added by new requirements or missed during initial development of the code.

If this research could have shown that defects correlated with cyclomatic complexity, the information could be used in conjunction with these classes of fixes to predict possible defects in the code.

### *4.6.2 Comparing Absolute Change in Size with Affected Statements*

This correlation was computed to see if there was much difference between the total statements affected (added, changed, or deleted) of a module and the absolute change in the module's size. This correlation would reveal if program changes were primarily comprised of modifications to existing code versus adding and deleting lines. Modifying existing statements would not affect the change in size of the module; however, it would increase the number of affected statements.

The results show a very high, positive correlation between affected statements and the size change of the module. This would suggest, as previously stated, that most changes involved adding or deleting statements instead of modifying existing statements. Referring to the previous comparison of change in size to change in cyclomatic complexity, it was suggested that the classes of fixes observed addressed new exceptions. This would generally cause additional code to be added versus changing the existing code thus causing the size change and affected statements to relatively equal.

### *4.6.3 Comparing Absolute Change in Cyclomatic Complexity with Affected Statements*

This correlation is much similar to the correlation computed in section 4.6.1 except that it looked at the affected statements instead of just the change in size, and the absolute change in cyclomatic complexity was used instead of only the change in cyclomatic complexity (could have been positive or negative). Due to the high correlation of size with affected statements and the fact that cyclomatic complexity correlated highly with change in size, it is intuitive that this correlation would also be quite strong. If the changes observed in this research had involved different classes of fixes that did not predominately add code, the results of this comparison could have been much different.

## *4.7 Summary*

This chapter discussed the findings to the comparisons made between the software measures described in Chapter 3 of this thesis. It described how the defect data provided by the owners of the system studied could not be used and therefore no comparisons could be made between the product measures and defect measures. However, several comparisons were done between manhours and the product measures gathered.

High correlations were found between manhours and the product measures for the new code that was developed in each revision. However, there were only 10 programs and due to the Central Limit Theorem [Dev95], the data could not be considered as a good representation of the population. No significant correlations were found between manhours and the product measures for the code that was modified in each revision.

This chapter also discussed the correlations that were found between changes in product measures between revisions. Suggestions were provided on what may have attributed to these correlations. Finally, the chapter discussed the correlations that were also found between the size and the complexity of the baseline code.

Chapter 5 will discuss some problems with the way process measures are being identified by the owners of the system that was studied. It will also provide suggestions on how these process measures should be identified to make them more useful to analyze software processes.

# 5. Conclusions and Recommendations

This chapter concludes this thesis by presenting some conclusions and recommendations. It covers some ideas and discusses what the findings may indicate. This chapter also provides recommendations on how data could be collected so better linear correlations could be attempted between measures. Finally, it provides suggestions for future research and closes with a few final remarks.

## 5.1 Conclusions to the Findings

Two of the initial hypotheses of this research were that defect data and manhour data would have a linear correlation with the software complexity. This turned out to be quite difficult to assess with the data provided. This was because the measurements provided by the organization under study were at a level that did not provide enough detail to conduct statistically sound analysis. Jones states [Jon91] that for measurements to be more useful, organizations should separate the data into meaningful subsets by breaking the measurements down into the smallest possible pieces. This researcher believes that if the processes measured were recorded at the lowest possible level, better analysis could have been conducted on the data, and results could have been more conclusive.

### 5.1.1 Improving Defect Data Collection

The defect data provided could not be used to attempt any comparisons with complexity due to the nature and mapping of the defects. Also, the only defects detected over the period of the three releases were found during the development or maintenance of the code; no defects were reported from the customer during this time frame. Although it is impressive that no

defects were found in the field during this period of time, it's not all that surprising. The system that was analyzed has been in the field for over 20 years. Due to its longevity, it is likely that most of the defects in the code have already surfaced and been eliminated. Also, the modifications to the code over the three releases were relatively small compared to the size of the system. This makes it less likely that errors would be introduced into the system.

To be more useful, defect information should be recorded at the lowest possible level in the software system. For example, if a defect is found during a development phase or from the field, attempts should first be made to map the defect to the part of statement that was the source of the defect. If this cannot be determined, attempt to map the defect to the statement that was the source of the defect. If a statement cannot be determined, map the defect to module. And finally, if none of these lower level mapping can be done, map the defect to the program that was the source of the defect. With these low-level mappings, better analysis can be conducted on the source code and defect data to see if trends exist for code that causes an inordinate amount of defects, or if certain program structures or attributes may cause an inordinate amount of defects.

Once a history of defect data is recorded at the lowest possible level over a period of at least 4 releases, this researcher would suggest reinvestigating the correlation between the defect data and the cyclomatic complexity of the code.

### *5.1.2 Improving Manhour Data Collection*

The manhour data provided by the sponsor mapped to the individual programs that were added or modified during the three revisions of the system. This allowed correlations to be computed between manhours and the cumulative complexity and size measures of the program.

The highest sample correlation coefficient found was between manhours and affected statements during a change. However, as discussed in Chapter 4, the overall comparison of manhours with affected statements was significantly influenced by the new code.

Although the new code sample set was small, it was noted that tracking manhours with new code is more reflective of the work being done. It is difficult to say what could attribute to the low degree of linear correlation between manhours and the changed code. Several factors could have influenced the time it took to complete the changes such as complexity of the code, determining how to extend the existing code, or cross-team communications, etc.

The manhour data could be more useful if it were broken down into smaller subsets such as the time needed for requirements, analysis, design, and coding, etc. This type of low-level recording could provide insight into why some sections of code took longer to code than other sections of code. Trends could be investigated at lower levels to determine if size or complexity plays a factor in the time it takes to: understand requirements, analyze the code, design the code, or write the code.

Once a history of manhour data is recorded at a lower level, the data could be analyzed to determine which areas of the development process are taking the most time. The low-level data could help identify and eliminate bottlenecks in the development process. It could also identify areas of the development process that need defined processes.

## *5.2 Suggestions for Future Research*

The system analyzed in this research has been in the field for over 20 years. With a system this far into its lifecycle, it is not surprising that the modifications being made to it are small, and the amount of defects being discovered by customers is negligible. This could be because the system has matured and no longer requires many changes. The low amount of

defects being found in the field could indicate that the system has been in existence for so long that most of the defects have been found and corrected.

Future research should attempt to find a system early in its lifecycle: perhaps only a few years old. This type of system could be going through larger modifications and experiencing more field defects than systems that have been in the field awhile. Another factor to consider is the maturity of the organization's software process. Finding an organization that maintains low-level measurements for their software processes could ensure the data being used for the analysis is more valid [Jon91].

If a history of low-level process measures are available for future research, the primary comparison this researcher would suggest reinvestigating is the defect data versus cyclomatic complexity. If these two measures showed a strong correlation, the information could help software managers estimate defects by evaluating the source code. With this information, managers could predict potential problems that could cause schedule slippages and cost overruns. Software managers could also use the information to modify processes to ensure the cyclomatic complexity of modules are kept at the lowest level possible, thus reducing the possibility of defects. Another comparison this researcher suggests reinvestigating is the manhour data versus cyclomatic complexity. For many of the same reasons stated above, if correlations can be found between manhour data and cyclomatic complexity, the information could be used to help software managers directly estimate manhours by estimating the cyclomatic complexity from specifications and requirements of future projects of the same nature.

For future studies, this researcher suggests not using Henry and Kafura's Information Flow measure for analyzing COBOL programs. The Information Flow measure is highly

influenced by the size of the module being measured. Often, COBOL modules contain many statements which create a large variance in the results of the measure from one module to the next. Another factor to consider is that only global variables are used for COBOL systems. COBOL modules do not pass parameters between each other when they are called. This eliminates a large portion of what the metric is trying to measure. And finally, COBOL modules update and use many variables. In this research, there were so many variables being used and updated in some modules that the information flow measure returned numbers in the billions. This made the measure practically useless due to the enormous variance between modules.

### *5.3 Contributions*

This research provided empirical data to the relatively unexplored field of using metrics on COBOL systems. Considering that COBOL is still the predominant programming language worldwide, this researcher feels more work needs to be done with metrics in the COBOL arena. This research also identified weaknesses in the existing metrics effort used by the owners of the system being studied. It also provided suggestions on how software organizations can collect more meaningful measures on their software. It showed the importance of collecting data at low levels and how the low-level data could possibly be used to help software managers improve processes and predict other lifecycle activities. If measures are recorded at the lowest level possible, they provide a sound, empirical base on which comparisons can be measured. Recording measures at the lowest possible level also eliminates errors that could be made when interpreting measures that were recorded at a higher level.

## *5.4 Final Remarks*

It is likely that product measures and metrics derived from source code can play a significant role in helping software managers and software engineers improve the way they develop software systems. Although this research was inconclusive in determining that source code measures can be used to estimate other process measures, this researcher believes, given a COBOL system that is relatively young with a history of maintaining low-level process measures, that product measures such as cyclomatic complexity could be used to estimate other lifecycle process measures on COBOL legacy systems.

# *Bibliography*


Ada92 AdaMAT™ *Reference Manual, Version 1.0.* Andover, Massachusetts. Dynamics Research Corporation Systems Division

Ana96 Analytical Software. *Statistix® for Windows User's Manual.* Tallahassee Florida: Analytical Software, 1996.

Bau92 Baumert, John H., McWhinney, Mark S. "Software Measure and the Capability Maturity Model", *Technical Report CMU/SEI-92-TR-25.* Pittsburgh, Pennsylvania: Software Engineering Institute, Carnegie Mellon University, September 1992.

Bro95 Brooks, Frederick P. Jr. *The Mythical Man-Month: Essays on Software Engineering.* Menlo Park, California: Addison-Wesley Longman, Inc., 1995.

Con86 Conte, S. D., Dunsmore, H.E., & Shen V.Y. *Software Engineering Metrics and Models.* Menlo Park California: The Benjamin/Cummings Publishing Co. Inc., 1986.

Cur79 Curtis, B., S. Sheppard, P. Hilliman, M. Borst, and T. Love "Measuring the Psychological Complexity of Software Maintenance Tasks with the Halstead and McCabe Metrics," *IEEE Transactions on Software Engineering, SE-5, 2*: 96-104 (March 1979).

Dev95 Devore, Jay L. *Probability and Statistics for Engineering and the Sciences, 4th ed.* New York: Wadsworth Inc., 1995.

Fen97 Fenton, Norman E. & Shari Lawrence Pfleeger. *Software Metrics: A Rigorous Approach.* London: PWS Publishing Co., 1997.

Fens97 Fenstermacher, Scott. McCabe & Associates. Electronic Mail Correspondence. Address: scott@mccabe.com. 7 August 1997

Gib89 Gibson, V. and J. Senn, "System Structure and Software Maintenance Performance," *Communications of the ACM,. 32, 3*: 347-358 (March 1989)

Gra94 Grauer, Robert T. and Carol Vazquez Villar. *COBOL From Micro to Mainframe.* Englewood Cliffs NJ: Prentice Hall, Inc., 1994.

Hal77 Halstead, Maurice H. *Elements of Software Science.* New York: Elsevier North-Holland, Inc., 1977.

Hei94 Heimann, David I., "Using Complexity-Tracking in the Software Development Process," *Proceedings of the 1994 McCabe Users Group Conference, Section III*: 1-36 (May 1994)

Hen81 Henry, Sallie and Dennis Kafura. "Software Structure Metrics Based on Information

Flow," *IEEE Transactions on Software Engineering, SE-7, 5:* 510-517 (September 1981).

Hen81b Henry, S., D. Kafura, and K. Harris. "On the Relationships Among Three Software Metrics," *1981 ACM Workshop/Symposium on Measurement and Evaluation of Software Quality*, (March 1981)

Hen79 Henry, Sallie Marie *Information Flow Metrics for the Evaluation of Operating Systems' Structure*. Ph.D. dissertation. Iowa State University, Ames Iowa, 1979.

Hum89 Humphrey, Watts S. *Managing the Software Process*. New York: Addison-Wesley Publishing Co., 1989.

Hum92 Humphrey, Watts S. "Introduction to Software Process Improvement," *Technical Report CMU/SEI-92-TR-7*. Pittsburgh, Pennsylvania: Software Engineering Institute, Carnegie Mellon University, June 1992.

Jon91 Jones, Capers. *Applied Software Measurement: Assuring Productivity and Quality*. New York: McGraw-Hill, Inc. 1991.

Kan93 Kanko, Mark A. "On Gas Gauges and Software Metrics" *Cross Talk: The Software Engineering Report*: 9-18 (Special Edition 1993).

McC76 McCabe, Thomas J. "A Complexity Measure," *IEEE Transactions on Software Engineering, SE-2, 4*: 308-321 (December 1976).

McC96 McCabe, Thomas. "Cyclomatic Complexity and the Year 2000," *IEEE Software*: 115-117 (May 1996).

Mil94 Department of Defense. *Software Development and Documentation*. MIL-STD-498. Washington: GPO, 5 December 1994.

Moe91 Moe, Michelle. *Control-flow complexity metrics for COBOL programs*. MS thesis, South West Texas State University, San Marcos Texas, 1991.

Nic86 Nickerson, Robert C. *Fundamentals of Structured COBOL*. Boston, Massachusetts: Scott, Foresman and Company., 1986.

Pau93 Paulk, Mark C., Bill Curtis, Mary Beth Chrissis, Charles V. Weber. *Capability Maturity Model for Software, Version 1.1*. Pittsburgh, Pennsylvania: Software Engineering Institute, February 1993.

Pin97 Pine, Jim. *Headquarters On-Line System for Transportation (HOST) Program Management Review*. Computer Systems Squadron, Air Mobility Command, Scott AFB, IL, 10 Jun 1997.

Rea90 Reasoning Systems. *Refine Users Guide*. Palo Alto California: Reasoning Systems

Inc., May 1990.

Rea95 Reasoning Systems. *Refine/COBOL Users Guide*. Palo Alto California: Reasoning Systems Inc., May 1995.

Rum91 Rumbaugh, James, Michael Blaha, William Premerlani, Frederick Eddy, and William Lorensen. *Object-Oriented Modeling and Design*. Englewood Cliffs, New Jersey: Prentice Hall Inc. 1991.

Sch79 Schneidewind, N. and H. Hoffmann, "An Experiment in Software Error Data Collection and Analysis," *IEEE Transactions on Software Engineering, SE-5, 3*: 276-286 (May 1979)

She88 Shepperd, Martin. "An evaluation of Software Product Metrics," *Information and Software Technology, 30*: 177-188 (April 1988).

Shep88 Shepperd, Martin. "A Critique of Cyclomatic Complexity as a Software Metric," *Software Engineering Journal*: 30-36 (March 1988).

She92 Shepperd, Martin. "Products, processes and metrics," *Information and Software Technology, 34*: 674-680 (October 1992).

Tri95 Tristram, Claire, "Do you really (still) need...COBOL?," *Open Computing*: 57-58 (June 1995).

Wal79 Walsh, T., "A Software Reliability Study Using a Complexity Measure," *AFIPS Conference Proceedings, National Computer Conference, 48*: 761-768, 1979.

Wal96 Wallace, Dolores R., Arthur H. Watson, Thomas J. McCabe. "Structured Testing: A Testing Methodology Using the Cyclomatic Complexity Metric," *National Institute of Standards and Technology Special Publication 500-235*: (August 1996).

War89 Ward, W., "Software Defect Prevention Using McCabe's Complexity Metric," *Hewlett-Packard Journal, 40, 2*: 64-69 (April 1989)

# *VITA*

Lieutenant Richard Edward Boone was born on 1 May 1962 in Monroe, Michigan. He spent his entire childhood in Southeastern Michigan where he graduated in 1980 from Airport High School in the small village of Carleton. In October, 1981, he enlisted in the Air Force as a Telecommunications Operator. He later became a computer operator, then a computer programmer.

He spent a little over 12 years enlisted attaining the rank of Technical Sergeant. His assignments included Rockville, Iceland; Patrick AFB, Florida; Hahn AB, Germany; Camp Smith, Hawaii; and the U.S. Air Force Academy, Colorado respectively. While stationed in Hawaii, he began attending school part time at Hawaii Pacific University (HPU). He continued taking classes at local colleges after moving to Colorado and graduated Cum Laude from HPU in June, 1992 with a Bachelor of Science Degree in Computer Science. He later applied for Officer's Training School (OTS) and was accepted.

After graduating OTS in March, 1994, he was assigned to the Reconnaissance Systems Program Office, Aeronautical Systems Center, Wright-Patterson AFB where he worked as a Acquisitions Program Manager and Executive Officer.

In May, 1996, he entered the School of Engineering, Air Force Institute of Technology (AFIT). He graduated on 16 Dec 1997, earning a Master of Science in Computer Systems, with a specialization in Software Engineering.

Following AFIT, he was assigned to the HQ Materiel System Group, Hill AFB, Utah.

Permanent Address: 13809 Sumpter Road
Carleton, Michigan 48117

# *Appendix A*

*Refine/Cobol* Code

Table of Contents

Program for Collecting Product Measures

```
% Cob-out.re - code for computing the cobol source code
%  product metrics.  It computes the following metrics:
% - Size of a module
% - McCabe's Cyclomatic Complexity Number of a module
% - Maximum Nesting Level of a module
% - Information Flow of a module
%  09/09/97

%  The following is needed to include the Cobol features
%  of Refine/Cobol

!! in-package("RCBU")
!! in-grammar('user)


var seq-secs: seq(procedure-section) = []          % A sequence of sections in a procedure
var seq-paras: seq(procedure-paragraph) = []       % A sequence of paragraphs in a section
var seq-ifs: seq(if-statement) = []                % Used for nested if statements
var seq-case-ifs: seq(if-statement) = []           % Used for nested ifs in a case statement
var temp-seq: seq(if-statement) = []               % Temporary variable for nested if check
var para-name: string = []                         % Paragraph Name
var prog-name: string = []                         % Program Identification
var num-of-paras: integer = 0                      % Number of paragraphs in the program
var max-nest-lvl: integer = 1                      % The highest nesting level in the program
var nest-lvl: integer = 1                          % The current statement's nesting level
var para-lvl: integer = 1                          % The highest nesting level in a paragraph
var para-cyclo: integer = 0                        % Cyclomatic number of the paragraph
var max-para-cyclo: integer = 0                    % Program's highest cyclomatic paragraph
var min-para-cyclo: integer = 99                   % Program's lowest cyclomatic paragraph
var max-cyclo-para-names: seq(string) = []         % Names of highest cyclomatic paragraph
var max-info-flow-para-names: seq(string) = []     % Names of highest info flow  paragraph
var min-cyclo-para-names: seq(string) = []         % Names of lowest cyclomatic paragraph
var max-nest-para-names: seq(string) = []          % Names of highest nesting paragraph
var seq-in-var: seq(identifier-ref) = []           % Sequence of variables used in a paragraph
var seq-out-var: seq(identifier-ref) = []          % Sequence of variables updated in a para
var para-fan-in: integer = 0                       % The fan-in of a paragraph
var para-fan-out: integer = 0                      % The fan-out of a pragraph
var para-vars-used: integer = 0                    % Num of variables used in a paragraph
var para-vars-updated: integer = 0                 % Num of variables updated in a paragraph
var para-num-of-statements: integer = 0            % Num of statements (length) in a para
var para-info-flow: integer = 0                    % Information Flow number for a para
var max-para-info-flow: integer = 0                % Highest Info flow in the program
var in-term-count: integer = 0                     % Counter for used variables
```

```
var out-term-count: integer = 0                  % counter for updated variables
var proc-div: procedure-division = undefined     % The entire procedure division
var para-struct-name: symbol = undefined         % Name of a paragraph
var p-name: symbol = undefined                   % Name of current paragraph
var para-calls-to-para: integer = 0              % Number of calls to current paragraph
var total-calls-from: integer = 0                % Number of calls from current paragraph

% The following function is the driver of this program. It takes a sequence of files
% already loaded in the varible seq-file which is done in another program and analyzes
% each file at a time

function domore ( ) =
  (enumerate myfile over seq-file do
    doseq(file-definitions(myfile)))

% The following function first gets the programs identification, then it processes the
% procedure division.  Once the procedure division is processed, it prints out the
% statistics for the program

function doseq(cobseq: seq(program)) =
  (enumerate prog over cobseq do
     program-identification-division(prog) - - >
          get-prog-name(program-identification-division(prog));
     program-procedure-division(prog) - - >
          do-proc-div(program-procedure-division(prog)));
   reset-vars ( )

% The following function simply resets all the variables after each file is analyzed

function reset-vars ( ) =
  seq-ifs <- [ ];
  seq-case-ifs <- [ ];
  temp-seq <- [ ];
  num-of-paras <- 0;
  seq-paras <- [ ];
  seq-secs <- [ ];
  para-name <- [ ];
  prog-name <- [ ];
  num-of-paras <- 0;
  max-nest-lvl <- 1;
  nest-lvl <- 1;
  para-lvl <- 1;
  para-cyclo <- 0;
  max-para-cyclo <- 0;
  min-para-cyclo <- 99;
```

```
max-cyclo-para-names <- [];
max-info-flow-para-names <- [];
min-cyclo-para-names <- [];
max-nest-para-names <- [];
seq-in-var <- [];
seq-out-var <- [];
para-fan-in <- 0;
para-fan-out <- 0;
para-vars-used <- 0;
para-vars-updated <- 0;
para-num-of-statements <- 0;
para-info-flow <- 0;
max-para-info-flow <- 0;
in-term-count <- 0;
out-term-count <- 0;
para-calls-to-para <- 0;
total-calls-from <- 0;
proc-div <- undefined;
NIL

% The following function gets the programs name from the identification division

function get-prog-name(id-div: identification-division) =
      prog-name <- cdif-di-to-string(program-name(id-div))

% The following function checks to see if there is only one or several sections in the
% procedure division.  If the program has only one section, it will be converted to a
% sequence of sections (with only one element) so a common procedure can be called to
% process the procedure division.

function do-proc-div(proc: procedure-division) =
     proc-div <- proc;
     defined?(procedure-division-initial-section(proc)) - - >
        do-secs([procedure-division-initial-section(proc)]);
     defined?(procedure-division-section-sequence(proc)) - - >
        do-secs(procedure-division-section-sequence(proc))

% The following function processes the section(s) of the procedure division.
```

```
% This checks to see if there is only one or several paragraphs in the section.
% If the section has only one paragraph, it will be converted to a sequence of
% paragraphs (with only one element) a common procedure can be called to process
% the section.

function do-secs(secs: seq(procedure-section)) =
    (enumerate sec over secs do
     defined?(procedure-section-initial-paragraph(sec)) - - >
       do-paras([procedure-section-initial-paragraph(sec)]);
     defined?(procedure-section-paragraph-sequence(sec)) - - >
       do-paras(procedure-section-paragraph-sequence(sec)))

% The following function process the individual paragraphs in a section. It first gets
% the name of the paragraph and then continues by processing each sentence of the
% paragraph

function do-paras(paras: seq(procedure-paragraph)) =
  (enumerate para over paras do
   (para-struct-name <- undefined;
    para-name <- "MAIN";
    defined?(procedure-paragraph-name(para))
        - - > (para-struct-name <- identifier-name(procedure-paragraph-name(para));
               para-name <- cdif-di-to-string(procedure-paragraph-name(para));
               size(para-name) > 2 - - > (para-name <- delete(para-name, 1);
               para-name <- delete(para-name, 1)));
    reset-para-vars ( ) ;
    check-for-calls-to ( ) ;
    count-calls-from(para);
    count-num-of-statements(para);
    do-sentences(procedure-paragraph-sentence-sequence(para));
    get-stats ( ) ;
    print-para-stats ( ) ))


% The following function resets the paragraph variable counters each time
% a paragraph is about to be processed

function reset-para-vars ( ) =
    para-calls-to-para <- 0;
    total-calls-from <- 0;
    para-cyclo <- 1;
    para-lvl <- 1;
    para-fan-in <- 0;
    para-fan-out <- 0;
    para-vars-used <- 0;
```

```
    para-vars-updated <- 0;
    para-info-flow <- 0


% The following function computes the paragraph's metrics & updates the overall
% program's metrics if necessary.

function get-stats ( ) =
    num-of-paras <- num-of-paras + 1;
    para-vars-used <- size(seq-in-var);
    para-vars-updated <- size(seq-out-var);
    para-fan-in <- para-calls-to-para + para-vars-used;
    para-fan-out <- total-calls-from + para-vars-updated;
    para-info-flow <- para-num-of-statements *
                ((para-fan-in * para-fan-out) *
                 (para-fan-in * para-fan-out));
    seq-in-var <- [];
    seq-out-var <- [];
    para-cyclo > max-para-cyclo -->
      (max-cyclo-para-names <- [];
       max-para-cyclo <- para-cyclo;
       max-cyclo-para-names <- append(max-cyclo-para-names, para-name));
    (para-cyclo = max-para-cyclo AND
     para-name ~in max-cyclo-para-names) -->
       max-cyclo-para-names <- append(max-cyclo-para-names, para-name);
    para-cyclo < min-para-cyclo -->
      (min-cyclo-para-names <- [];
       min-para-cyclo <- para-cyclo;
       min-cyclo-para-names <- append(min-cyclo-para-names, para-name));
    (para-cyclo = min-para-cyclo AND
     para-name ~in min-cyclo-para-names) -->
       min-cyclo-para-names <- append(min-cyclo-para-names, para-name);
    para-info-flow > max-para-info-flow -->
      (max-info-flow-para-names <- [];
       max-para-info-flow <- para-info-flow;
       max-info-flow-para-names <- append(max-info-flow-para-names, para-name));
    (para-info-flow = max-para-info-flow AND
     para-name ~in max-info-flow-para-names) -->
       max-info-flow-para-names <- append(max-info-flow-para-names, para-name);
     para-num-of-statements = 0 --> (para-cyclo <- 0;
                         para-lvl <- 0;
                         para-info-flow <- 0)
```

```
% The following function prints out the current paragraph's
% statistics/metrics to include the sub-system name, program name, paragraph-name
% number of statements in the paragraph, the McCabe's number for the paragraph,
% the maximum nesting level in the paragraph, and the information flow metric for
% thr paragraph

function print-para-stats ( ) =
      format(true,"~%~D ~D ~D ~D ~D ~D ~D",
         sub-name,prog-name,para-name,para-num-of-statements,
          para-cyclo,para-lvl,para-info-flow)


% The following function enumerates over the sequence of all sentences in a paragraph
% and analyzes them

function do-sentences(sentences: seq(procedure-sentence)) =
    (enumerate sent over sentences do
       do-statements(procedure-sentence-statement-sequence(sent)))

% The following function checks to see if the sentences contribute to cyclomatic
% complexity if the sentence is an if-statement, case-statement, or perform it may
% increase the cyclomatic complexity.  This function also calls a function that will
% compute the nesting level of an if-statement%

function do-statements(statements: seq(statement)) =
    (enumerate state over statements do
      do-control-flow(state);
      if-statement(state) - - >
            (do-if-cyclo(set-to-seq(descendants(state)));
             nest-lvl < - 2;
             check-nesting(state);
             check-max-nesting ( ) );
      evaluate-statement(state) - - >
             (nest-lvl < - 2;
             (enumerate whn over filter(lambda (obj: object)
                                evaluate-case(obj),
              set-to-seq(kids(state))) do
               seq-case-ifs < -
                  concat(seq-case-ifs,
                    set-to-seq(filter(lambda (obj: object)
                                 if-statement(obj),
                                      kids(whn)))));
             size(seq-case-ifs) > 0 - - > check-level(seq-case-ifs);
             do-case-cyclo(set-to-seq(descendants(state)));
             check-max-nesting ( ) );
      perform-statement(state) - - >
```

```
            do-perf-cyclo(state);
        read-statement(state) - - > do-read-write-cyclo(state);
        search-statement(state) - - > do-search-cyclo(state);
        write-statement(state) - - > do-read-write-cyclo(state);
        return-statement(state) - - > do-return-cyclo(state))

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%  THIS SECTION OF CODE COMPUTES SIZE OF THE CURRENT MODULE
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% The following function counts the number of statements in a paragraph
% the result is used as the length portion to compute the Information
% flow metric for each paragraph

function count-num-of-statements(para: paragraph) =
   para-num-of-statements <- size(filter(lambda (obj: object) statement(obj),
           set-to-seq(descendants(para))))


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%  THIS SECTION OF CODE COMPUTES MCCABE'S CYCLOMATIC COMPLEXITY NUMBER
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% The following function processes a perform statement, it looks to see if it
% is a "perform until" or "perform varying until" statement because these are
% loop structures in Cobol and contribute to the cyclomatic complexity

function do-perf-cyclo(perf: perform-statement) =
 defined?(perform-statement-until-clause(perf)) - - >
   (para-cyclo <- para-cyclo + 1;
   get-perf-cond(perform-statement-until-clause(perf)));
 defined?(perform-statement-varying-clause(perf)) - - >
   (para-cyclo <- para-cyclo + 1;
   (defined?(perform-object-until-clause
         (perform-statement-varying-clause(perf))) - - >
    get-perf-cond(perform-object-until-clause
           (perform-statement-varying-clause(perf)))))
```

```
% The following function calls another function that process the conditions of
% the until portion of the perform statement

function get-perf-cond(uclause: perform-until-clause) =
        count-perf-cond(perform-object-condition-expression(uclause))

% The following function checks to see if the conditons of the perform until
% are compound, if so it increases the cyclomatic complexity with each
% compound conjunction

function count-perf-cond(exp: condition-expression) =
      (enumerate e over set-to-seq(descendants(exp)) do
       (or-condition(e)) AND NOT
       (relation-condition(e)) AND NOT
       (simple-condition(e)) AND NOT
       (not-condition(e)) - - > (para-cyclo < - para-cyclo + 1))

% The following function processes an if statement. It counts all nested if statements
% and compound conditions of each if statement which increases the overall cyclomatic
% complexity of the statement by one. It also checks any nested perform statements to
% see if they are loops

function do-if-cyclo(seqnodes: seq(statement)) =
     (enumerate s over seqnodes do
       if-statement(s) - - > (para-cyclo < - para-cyclo + 1);
       (or-condition(s)) AND NOT
       (relation-condition(s)) AND NOT
       (simple-condition(s)) AND NOT
       (not-condition(s)) - - > (para-cyclo < - para-cyclo + 1);
       perform-statement(s) - - > do-perf-cyclo(s))

%  This function checks any read or write statements to see if they contribute to
%  the cyclomatic complexity. Read or Write statements may include a looping
%  condidition which would add to the cyclomatic complexity

function do-read-write-cyclo(state: statement) =
   para-cyclo < - para-cyclo + 1;
   defined?(proc-div-obj-on-condition-clause(state)) - - >
      do-statements(on-condition-clause-statement-sequence
                  (proc-div-obj-on-condition-clause(state)));
   defined?(proc-div-obj-not-on-condition-clause(state)) - - >
      do-statements(on-condition-clause-statement-sequence
               (not-clause-on-condition-clause
                  (proc-div-obj-not-on-condition-clause(state))))
```

```
% This function checks any search statements to see if they contribute to
% the cyclomatic complexity. Search statements may include a looping
% condidition which would add to the cyclomatic complexity

function do-search-cyclo(state: statement) =
  para-cyclo <- para-cyclo + 1;
  defined?(proc-div-obj-on-condition-clause(state)) - - >
    do-statements(on-condition-clause-statement-sequence
                  (proc-div-obj-on-condition-clause(state)));
  defined?(search-statement-when-clause-sequence(state)) - - >
   (enumerate w over search-statement-when-clause-sequence(state) do
    (para-cyclo <- para-cyclo + 1;
     do-statements(search-when-clause-statement-sequence(w));
     (enumerate cond over
              set-to-seq(descendants
                    (search-when-clause-condition-expression(w))) do
       (or-condition(cond)) AND NOT
       (relation-condition(cond)) AND NOT
       (simple-condition(cond)) AND NOT
       (not-condition(cond)) - - > para-cyclo <- para-cyclo + 1)))

% The following function processes a case statement. Each case alternative increases
% the cyclomatic complexity of the structure by one. It will also check to see if
% any of the alternatives contain perform statements that contribute to the overall
% cyclomatic complexity.

function do-case-cyclo(seqnodes: seq(object)) =
    (enumerate s over seqnodes do
     evaluate-case(s) - - > para-cyclo <- para-cyclo + 1;
     perform-statement(s) - - > do-perf-cyclo(s);
     if-statement(s) - - > do-if-cyclo(set-to-seq(descendants(s))))

% This function checks any return statements to see if they contribute to
% the cyclomatic complexity. Return statements may include a looping
% condidition which would add to the cyclomatic complexity

function do-return-cyclo(state: statement) =
    defined?(proc-div-obj-on-condition-clause(state)) - - >
     do-statements(on-condition-clause-statement-sequence
                  (proc-div-obj-on-condition-clause(state)));
    defined?(proc-div-obj-not-on-condition-clause(state)) - - >
     do-statements(on-condition-clause-statement-sequence
                (not-clause-on-condition-clause
                  (proc-div-obj-not-on-condition-clause(state))))
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%  THIS SECTION OF CODE COMPUTES MAXIMUM NESTING LEVEL OF THE MODULE
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% The following function takes the kid of an if-statement and filters out only
% the kids that are also if-statements.  It then passes this to another function
% for additonal nesting processing

function check-nesting(stmt: statement) =
    seq-ifs <-  filter(lambda (obj: object) If-statement(obj),
            set-to-seq(Kids(stmt)));
    check-level(seq-ifs)

% The following function takes the kids of an if statement and recursively gets
% their kids.  Each time an if-statement has kids that are also if-statement,
% it increases the nesting level by one

function check-level(seq-ifs: seq(if-statement)) =
    temp-seq <- [];
    size(seq-ifs) > 0  --> (nest-lvl <- nest-lvl + 1;
                  (enumerate s over seq-ifs do
                   (enumerate k over filter(lambda (obj: object)
                                  If-statement(obj),
                                   set-to-seq(Kids(s))) do
                      temp-seq <- append(temp-seq, k)));
                  check-level(temp-seq))

% This function checks the maximum nesting level in a paragraph.  It will check
% and see if the current nested  statement being checked
% has a higher level the current max level statement

function check-max-nesting () =
    nest-lvl > para-lvl --> para-lvl <- nest-lvl;
    nest-lvl > max-nest-lvl -->
                (max-nest-para-names <- [];
                 max-nest-lvl <- nest-lvl;
                 max-nest-para-names <- append(max-nest-para-names,
                                  para-name));
    nest-lvl =  max-nest-lvl AND
            para-name ~in max-nest-para-names -->
                 max-nest-para-names <- append(max-nest-para-names,
                                    para-name)
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%  THIS SECTION COMPUTES HENRY AND KAFURA'S INFORMATION FLOW METRIC
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% The following function checks the whole program to see how many
% other paragraphs call the current paragraph.  IT specifically looks for
% any PERFORM statements to the current paragraph.  This is used
% for the INformation Flow Metric

function check-for-calls-to ( ) =
 (enumerate perf over filter(lambda (obj: object) perform-statement(obj),
   set-to-seq(descendants(proc-div))) do
     (count-calls(identifier-name(procedure-ref-name
              (perform-statement-procedure-name(perf))));
           defined?(perform-statement-thru-procedure-name(perf)) - - >
              count-calls(identifier-name(procedure-ref-name
                       (perform-statement-thru-procedure-name(perf))))))

% The following function is a continuation of the above function, it keeps
% a tally of the number of calls to the current paragraph

function count-calls(pname: symbol) =
    pname = para-struct-name - - >
              para-calls-to-para < - para-calls-to-para + 1

% The following function counts the number of times the current paragraph
% calls other paragraphs/programs.  This is used for the Information Flow
% metric

function count-calls-from(para: paragraph) =
    total-calls-from < - size(filter(lambda (obj: object) perform-statement(obj),
                 set-to-seq(descendants(para))))
                + size(filter(lambda (obj: object) call-statement(obj),
                 set-to-seq(descendants(para))))

% The following function is the driver to get information that will
% be used to compute the INformation flow metric

function do-control-flow(state: statement) =
     if-statement(state) - - > do-control-flow-if(state);
     evaluate-statement(state) - - > do-control-flow-evaluate(state);
     do-control-flow-more(state)
```

```
% The following function is a contiuation of the above function
% it also gets information that will
% be used to compute the INformation flow metric

function do-control-flow-more(state: statement) =
     call-statement(state)  - - > do-control-flow-call(state);
     add-statement(state) - - > do-control-flow-add(state);
     subtract-statement(state) - - > do-control-flow-subtract(state);
     compute-statement(state) - - > do-control-flow-compute(state);
     accept-statement(state) - - > do-control-flow-accept(state);
     divide-statement(state) - - > do-control-flow-divide(state);
     multiply-statement(state) - - > do-control-flow-multiply(state);
     move-statement(state) - - > do-control-flow-move(state);
     set-statement(state) - - > do-control-flow-set(state);
     open-statement(state) - - > do-control-flow-open(state);
     close-statement(state) - - > do-control-flow-close(state);
     search-statement(state) - - > do-control-flow-search(state);
     initialize-statement(state) - - > do-control-flow-initialize(state);
     ((write-statement(state)) OR
      (read-statement(state))) - - > do-control-flow-read-write(state);
     sort-statement(state) - - > do-control-flow-sort(state);
      display-statement(state) - - > do-control-flow-display(state);
      perform-statement(state) - - > do-control-flow-perform(state);
      inspect-statement(state) - - > do-control-flow-inspect(state)

% The following function is used to gather info flow for
% an "inspect" statement

function do-control-flow-inspect(state: statement) =
   (enumerate insp over filter(lambda (obj: object) variable(obj),
            set-to-seq(descendants(state))) do
       check-the-in-seq(variable-name(insp)))

% The following function is used to gather info flow for
% a "display" statement

function do-control-flow-display(state: statement) =
    (enumerate v over
        filter(lambda (obj: object) variable(obj),
            set-to-seq(descendants(state))) do
             check-the-in-seq(variable-name(v)))
```

```
% The following function is used to gather info flow for
% a "perform" statement

function do-control-flow-perform(state: statement) =
      defined?(perform-statement-varying-clause(state)) - - >
        ((enumerate v-var over filter(lambda (obj: object) variable(obj),
                  set-to-seq(kids(perform-statement-varying-clause(state)))) do
                check-the-out-seq(variable-name(v-var)));
         defined?(perform-object-until-clause
                (perform-statement-varying-clause(state))) - - >
           (enumerate uv-var over filter(lambda (obj: object) variable(obj),
            set-to-seq(descendants(perform-object-until-clause
                      (perform-statement-varying-clause(state))))) do
                 check-the-in-seq(variable-name(uv-var))));
      defined?(perform-statement-until-clause(state)) - - >
        (enumerate u-var over filter(lambda (obj: object) variable(obj),
          set-to-seq(descendants(perform-statement-until-clause(state)))) do
                check-the-in-seq(variable-name(u-var)))


% The following function is used to gather info flow for
% a "set" statement

function do-control-flow-set(state: statement) =
    (enumerate s over filter(lambda (obj: object) variable(obj),
       set-to-seq(descendants(state))) do
          check-the-out-seq(variable-name(s)))


% The following function is used to gather information flow information
% from a call statement

function do-control-flow-call(state: statement) =
   defined?(call-statement-call-parameter-sequence(state)) - - >
     ((enumerate c over call-statement-call-parameter-sequence(state) do
       (enumerate v over filter(lambda (obj: object) variable(obj),
                  set-to-seq(descendants(c))) do
                    check-the-in-seq(variable-name(v)))))

% The following function is used to gather information flow information
% from a add statement

function do-control-flow-add(state: statement) =
    (enumerate asdi over add-statement-data-item-sequence(state) do
     (enumerate add-this over filter(lambda (obj: object) variable(obj),
       set-to-seq(descendants(asdi))) do
            check-the-in-seq(variable-name(add-this))));
```

```
    (enumerate asav over add-statement-arithmetic-variable-sequence(state) do
     (enumerate add-to over filter(lambda (obj: object) variable(obj),
       set-to-seq(descendants(asav))) do
            check-the-out-seq(variable-name(add-to))))

% The following function is used to gather information flow information
% from a subtract statement

function do-control-flow-subtract(state: statement) =
    (enumerate ssdi over subtract-statement-data-item-sequence(state) do
     (enumerate sub-this over filter(lambda (obj: object) variable(obj),
       set-to-seq(descendants(ssdi))) do
            check-the-in-seq(variable-name(sub-this))));
    (enumerate ssav over subtract-statement-arithmetic-variable-sequence(state) do
     (enumerate sub-from over filter(lambda (obj: object) variable(obj),
       set-to-seq(descendants(ssav))) do
            check-the-out-seq(variable-name(sub-from))))

% The following function is used to gather information flow information
% from an if statement

function do-control-flow-if(state:statement) =
    (enumerate s over
      (filter(lambda (obj: object) if-statement(obj),
       set-to-seq(descendants(state)))) do
         (enumerate i over filter(lambda (obj: object) variable(obj),
            set-to-seq(descendants(if-statement-condition-expression(s)))) do
               check-the-in-seq(variable-name(i))));
    (enumerate st over set-to-seq(descendants(state)) do
     do-control-flow-more(st))

% The following function is used to gather information flow information
% from an evaluate statement

function do-control-flow-evaluate(state:statement) =
    (enumerate eval over
      (filter(lambda (obj: object) variable(obj),
       set-to-seq(descendants(evaluate-statement-expression-1(state))))) do
         check-the-in-seq(variable-name(eval)));
    (enumerate e-stmt over set-to-seq(descendants(state)) do
       do-control-flow-more(e-stmt))
```

```
% The following function is used to gather information flow information
% from a move statement

function do-control-flow-move(state: statement) =
    (enumerate mov over filter(lambda (obj: object) variable(obj),
      set-to-seq(descendants(move-statement-data-item(state)))) do
          check-the-in-seq(variable-name(mov)));
   (enumerate var1 over move-statement-variable-sequence(state) do
          check-the-out-seq(variable-name(var1)))

% The following function is used to gather information flow information
% from a compute statement

function do-control-flow-compute(state: statement) =
    (enumerate com over filter(lambda (obj: object) variable(obj),
      set-to-seq(descendants(compute-statement-arithmetic-expression(state)))) do
          check-the-in-seq(variable-name(com)));
   (enumerate csav over compute-statement-arithmetic-variable-sequence(state) do
    (enumerate comp-result over filter(lambda (obj: object) variable(obj),
      set-to-seq(descendants(csav))) do
          check-the-out-seq(variable-name(comp-result))))

% The following function is used to gather info flow for
% a "search" statement

function do-control-flow-search(state: statement) =
  check-the-in-seq(variable-name(search-statement-variable(state)));
  defined?(search-statement-when-clause-sequence(state)) - - >
   (enumerate whn over search-statement-when-clause-sequence(state) do
     (enumerate sq over search-when-clause-statement-sequence(whn) do
       do-control-flow(sq));
     (enumerate cond over filter(lambda (obj: object) variable(obj),
       set-to-seq(descendants(search-when-clause-condition-expression(whn)))) do
       check-the-in-seq(variable-name(cond))));
  defined?(proc-div-obj-on-condition-clause(state)) - - >
    (enumerate sts over on-condition-clause-statement-sequence
                (proc-div-obj-on-condition-clause(state)) do
      do-control-flow(sts))


% The following function is used to gather info flow for
% a "read" or "write" statement

function do-control-flow-read-write(state: statement) =
  read-statement(state) - - > (check-the-in-seq(read-statement-name(state));
                  (defined?(read-statement-read-record(state)) - - >
                    check-the-out-seq
```

```
                        (variable-name
                         (read-record-variable
                          (read-statement-read-record(state))));
  write-statement(state) - - > (check-the-out-seq
                        (variable-name
                          (write-statement-name(state)));
                     (defined?(write-statement-from-clause(state)) - - >
                        check-the-in-seq(variable-name
                             (write-from-clause-variable
                              (write-statement-from-clause
                               (state))))));
  defined?(proc-div-obj-on-condition-clause(state)) - - >
    (enumerate on-c over (on-condition-clause-statement-sequence
                  (proc-div-obj-on-condition-clause(state))) do
          do-control-flow(on-c));
  defined?(proc-div-obj-not-on-condition-clause(state)) - - >
    (enumerate not-c over (on-condition-clause-statement-sequence
              (not-clause-on-condition-clause
                  (proc-div-obj-not-on-condition-clause(state)))) do
          do-control-flow(not-c))

% The following function is used to gather info flow for
% a "return" statement

function do-control-flow-return(state: statement) =
    check-the-in-seq(return-statement-name(state));
    defined?(proc-div-obj-on-condition-clause(state)) - - >
     (enumerate on-r over on-condition-clause-statement-sequence
                  (proc-div-obj-on-condition-clause(state)) do
       do-control-flow(on-r));
    defined?(proc-div-obj-not-on-condition-clause(state)) - - >
     (enumerate non-r over on-condition-clause-statement-sequence
                (not-clause-on-condition-clause
                  (proc-div-obj-not-on-condition-clause(state))) do
       do-control-flow(non-r))

% The following function is used to gather info flow for
% a "sort" statement

function do-control-flow-sort(state: statement) =
  check-the-in-seq(sort-statement-name(state));
  defined?(sort-statement-on-clause-sequence(state)) - - >
    (enumerate ocs over sort-statement-on-clause-sequence(state) do
     (enumerate s-var over filter(lambda (obj: object) variable(obj),
          set-to-seq(descendants(ocs))) do
            check-the-in-seq(variable-name(s-var))));
  defined?(sort-statement-input-clause(state)) - - >
```

```
      (enumerate inp over filter(lambda (obj: object) identifier-ref(obj),
           set-to-seq(descendants(sort-statement-input-clause(state)))) do
        check-the-in-seq(inp));
    defined?(sort-statement-output-clause(state)) - - >
      (enumerate op over filter(lambda (obj: object) identifier-ref(obj),
           set-to-seq(descendants(sort-statement-output-clause(state)))) do
        check-the-out-seq(op))

% The following function is used to gather info flow for
% an "initialize" statement

function do-control-flow-initialize(state: statement) =
    (enumerate iv over filter(lambda (obj: object) identifier-ref(obj),
       set-to-seq(descendants(state))) do
        check-the-out-seq(iv))

% The following function is used to gather information flow information
% from an accept statement

function do-control-flow-accept(state: statement) =
    defined?(accept-statement-from-name(state)) - - >
     (enumerate ac over filter(lambda (obj: object) variable(obj),
       set-to-seq(descendants(accept-statement-from-name(state)))) do
          check-the-in-seq(variable-name(ac)));
    check-the-out-seq(variable-name(accept-statement-variable(state)))

function do-control-flow-open(state: statement) =
   (enumerate o over filter(lambda (obj: object) identifier-ref(obj),
       set-to-seq(descendants(state))) do
          check-the-in-seq(o))

% The following function is used to gather info flow for
% a "close" statement

function do-control-flow-close(state: statement) =
   (enumerate c over filter(lambda (obj: object) identifier-ref(obj),
       set-to-seq(descendants(state))) do
          check-the-in-seq(c))


% The following function is used to gather information flow information
% from a divide statement

function do-control-flow-divide(state: statement) =
    defined?(divide-statement-data-item(state)) - - >
     (enumerate dvd over filter(lambda (obj: object) variable(obj),
       set-to-seq(descendants(divide-statement-data-item(state)))) do
```

```
            check-the-in-seq(variable-name(dvd)));
      defined?(divide-statement-data-item-1(state)) - - >
       (enumerate dvd over filter(lambda (obj: object) variable(obj),
         set-to-seq(descendants(divide-statement-data-item-1(state)))) do
            check-the-in-seq(variable-name(dvd)));
      defined?(divide-statement-data-item-2(state)) - - >
       (enumerate dvd over filter(lambda (obj: object) variable(obj),
         set-to-seq(descendants(divide-statement-data-item-2(state)))) do
            check-the-in-seq(variable-name(dvd)));
     (enumerate dsav over divide-statement-arithmetic-variable-sequence(state) do
      (enumerate div-result over filter(lambda (obj: object) variable(obj),
        set-to-seq(descendants(dsav))) do
            check-the-out-seq(variable-name(div-result))))

% The following function is used to gather information flow information
% from a multiply statement

function do-control-flow-multiply(state: statement) =
     defined?(multiply-statement-data-item(state)) - - >
      (enumerate mult over filter(lambda (obj: object) variable(obj),
        set-to-seq(descendants(multiply-statement-data-item(state)))) do
            check-the-in-seq(variable-name(mult)));
     defined?(multiply-statement-data-item-1(state)) - - >
      (enumerate mult over filter(lambda (obj: object) variable(obj),
        set-to-seq(descendants(multiply-statement-data-item-1(state)))) do
            check-the-in-seq(variable-name(mult)));
     defined?(multiply-statement-data-item-2(state)) - - >
      (enumerate mult over filter(lambda (obj: object) variable(obj),
        set-to-seq(descendants(multiply-statement-data-item-2(state)))) do
            check-the-in-seq(variable-name(mult)));
     (enumerate msav over multiply-statement-arithmetic-variable-sequence(state) do
      (enumerate mult-result over filter(lambda (obj: object) variable(obj),
        set-to-seq(descendants(msav))) do
            check-the-out-seq(variable-name(mult-result))))

% The following function is used to check the used variables in a paragraph
% it uses a sequence to put the variables in.  A variable is only added to
% the sequence if it is not already in the sequence.  This prevents counting
% a variable more than once if it appears several times in the paragraph

function check-the-in-seq(var1: identifier-ref) =
     in-term-count < - 0;
 (enumerate in-var over seq-in-var do
       (term-equal?(var1, in-var) - - > in-term-count < - 1));
      in-term-count = 0 - - > seq-in-var < - append(seq-in-var, var1)

% The following function is used to check the updated variables in a paragraph
```

```
% it uses a sequence to put the variables in.  A variable is only added to
% the sequence if it is not already in the sequence.  This prevents counting
% a variable more than once if it appears several times in the paragraph

function check-the-out-seq(var1: identifier-ref) =
    out-term-count <- 0;
    (enumerate out-var over seq-out-var do
     (term-equal?(var1, out-var) - - > out-term-count <- 1));
    out-term-count = 0 - - > seq-out-var <- append(seq-out-var, var1)
```

```
!! in-package("RCBU")
!! in-grammar('user)

% host-driver.re is the driver program that launches the code that loads and analyzes all four versions
% of the cobol source code
% 08/20/97

var seq-file: seq(cobol-object) = []
var sub-name: string = []

function go-host () =
 go-host-base () ;
 go-host-oct () ;
 go-host-jan () ;
 go-host-mar () ;
 go-host-apr () ;
 NIL
```

```
!! in-package("RCBU")
!! in-grammar('user)

% host-driver-base.re is the driver program that loads and analzyes the files in
% the baseline release of the software
% 08/20/97

function go-host-base () =
 seq-file <- [];
 go-over-base ();
 dribble("host-bo.txt");
 sub-name <- "OVER";
 domore ();
 dribble ();
 seq-file <- [];
 go-crqs-base ();
 dribble("host-bc.txt");
 sub-name <- "CRQS";
 domore ();
 dribble ();
 seq-file <- [];
 go-trais-base ();
 dribble("host-bt.txt");
 sub-name <- "TRAIS";
 domore ();
 dribble ();
 seq-file <- [];
 go-update-base ();
 dribble("host-bu.txt");
 sub-name <- "UPDATE";
 domore ();
 dribble ();
 seq-file <- [];
 go-maca-base ();
 dribble("host-bm.txt");
 sub-name <- "MACA";
 domore ();
 dribble ();
 NIL
```

```
!! in-package("RCBU")
!! in-grammar('user)

% host-driver-oct.re is the driver program that loads and analzyes the files modified
% or created for the october release of the software
% 08/20/97

function go-host-oct () =
 seq-file <- [];
 go-trais-oct ();
 dribble("host-ot.txt");
 sub-name <- "TRAIS";
 domore ();
 dribble ();
 seq-file <- [];
 go-update-oct ();
 dribble("host-ou.txt");
 sub-name <- "UPDATE";
 domore ();
 dribble ();
 NIL
```

```
!! in-package("RCBU")
!! in-grammar('user)

% host-driver-jan.re %  is the driver program that loads and analzyes the files modified
% or created for the january release of the software
% 08/20/97

function go-host-jan ( ) =
 seq-file <- [ ] ;
 go-trais-jan ( ) ;
 dribble("host-jt.txt");
 sub-name <- "TRAIS";
 domore ( ) ;
 dribble ( ) ;
 seq-file <- [ ] ;
 go-update-jan ( ) ;
 dribble("host-ju.txt");
 sub-name <- "UPDATE";
 domore ( ) ;
 dribble ( ) ;
 NIL
```

```
!! in-package("RCBU")
!! in-grammar('user)

% host-driver-apr.re is the driver program that loads and analzyes the files modified
% or created for the april release of the software
% 08/20/97

function go-host-apr () =
 seq-file <- [];
 go-maca-apr ();
 dribble("host-am.txt");
 sub-name <- "MACA";
 domore ();
 dribble ();
 seq-file <- [];
 go-update-apr ();
 dribble("host-au.txt");
 sub-name <- "UPDATE";
 domore ();
 dribble ();
 seq-file <- [];
 go-trais-apr ();
 dribble("host-at.txt");
 sub-name <- "TRAIS";
 domore ();
 dribble ();
 seq-file <- [];
 go-crqs-apr ();
 dribble("host-ac.txt");
 sub-name <- "CRQS";
 domore ();
 dribble ();
 NIL
```

```
!! in-package("RCBU")
!! in-grammar('user)

% load-crqs-base.re - tool for loading baseline cobol programs
% for the CRQS subsystem
% 08/20/97

function go-crqs-base () =
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/crqs/baseline/lbaauwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/crqs/baseline/lbablwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/crqs/baseline/lbacrwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/crqs/baseline/lbamjwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/crqs/baseline/lbamnwqs.u"));
NIL
```

```
!! in-package("RCBU")
!! in-grammar('user)

% load-maca-base.re - tool for loading the baseline cobol programs
% for the maca subsystem
% 08/20/97

function go-maca-base () =
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lyg12wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lyg43wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lyg44wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygadwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygaowqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygbtwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygccwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygdgwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygedwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygetwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygfpwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lyginwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygmawqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygmowqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygotwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygrnwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygs4wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygs5wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygx2wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygx3wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygx4wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/lygx5wqs.u"));
NIL
```

```
!! in-package("RCBU")
!! in-grammar('user)

% load-over-base.re - tool for loading baseline cobol programs
% for the over subsystem
% 08/20/97


function go-over-base () =
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/over/baseline/lygoswqs.u"));
NIL
```

```
!! in-package("RCBU")
!! in-grammar('user)

% load-trais-base.re - tool for loading baseline cobol programs
% for the trais subsystem
% 08/20/97

function go-trais-base () =
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxa0wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxa1wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxadwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxb0wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxbawqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxbbwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxbcwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxbxwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxi0wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxi1wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxi2wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxi3wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxi4wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxi5wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxi6wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxi7wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxi8wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxi9wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxiawqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxibwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxicwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxidwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxl0wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxl1wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxl2wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxl4wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxl5wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxl7wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxl9wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxlbwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxldwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxlewqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxlfwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxlgwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxlhwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxlmwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxlnwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxlowqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxlpwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxlzwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxm1wqs.u"));
```

```
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxmawqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxmbwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxmdwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxmewqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxmfwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxmhwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxmiwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxmjwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxmpwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxmqwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxmrwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxmswqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxmtwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxp4wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxpawqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxpbwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxpcwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxpdwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxpewqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxpfwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxphwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxpjwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxpkwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxplwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxpmwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxpnwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxppwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxpqwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxprwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxpvwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxpwwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxpywqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxpzwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxr0wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxr1wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxr2wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxr3wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxr5wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxr6wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxr7wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxrawqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxrdwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxrewqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxrgwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxriwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxrkwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxrmwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxrpwqs.u"));
```

```
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxrqwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxrrwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxrswqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxruwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxrwwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxrzwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxs1wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxs2wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxs3wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxt0wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxt1wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxt2wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxt3wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxtywqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxtzwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxu1wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxu3wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxu9wqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxubwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxuewqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxufwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxugwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxuhwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxupwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/lnxuxwqs.u"));
seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/ltpfowqs.u"));
NIL
```

```
!! in-package("RCBU")
!! in-grammar('user)

% load-update-base.re - tool for loading baseline cobol programs
% for the update subsystem
% 08/20/97

var seq-file: seq(cobol-object) = []

function go-update-base () =
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lyg15wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygabwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygalwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygbrwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygcdwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygcpwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygctwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygdawqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygdcwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygddwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygdmwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygdrwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygfbwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lyglfwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygobwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygorwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygpawqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygpcwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygpdwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygpfwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygptwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygtfwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/lygucwqs.u"));
NIL
```

```
!! in-package("RCBU")
!! in-grammar('user)

% load-trais-oct.re - tool for loading the oct release of cobol programs
% for the trais subsystem
% 08/20/97

function go-trais-oct () =
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15oct96/lnxiewqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15oct96/lnxd0wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15oct96/lnxd1wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15oct96/lnxd2wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15oct96/lnxd3wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15oct96/lnxd4wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15oct96/lnxd5wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15oct96/lnxd7wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15oct96/lnxr7wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15oct96/lnxicwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15oct96/lnxidwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15oct96/lnxmhwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15oct96/lnxpmwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15oct96/lnxrgwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15oct96/lnxpywqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15oct96/lnxrpwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15oct96/lnxu1wqs.u"));
 NIL
```

```
!! in-package("RCBU")
!! in-grammar('user)

% load-update-oct.re - tool for loading the oct release of cobol programs
% for the update subsystem
% 08/20/97

function go-update-oct () =
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/15oct96/lyg15wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/15oct96/lygabwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/15oct96/lygalwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/15oct96/lygdmwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/15oct96/lyglfwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/15oct96/lygobwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/15oct96/lygorwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/15oct96/lygpfwqs.u"));
NIL
```

```
!! in-package("RCBU")
!! in-grammar('user)

% load-trais-jan.re - tool for loading the jan release of cobol programs
% for the trais subsystem
% 08/20/97

function go-trais-jan () =
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15jan97/lnxmbwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/15jan97/lnxifwqs.u"));
 NIL
```

```
!! in-package("RCBU")
!! in-grammar('user)

% load-update-jan.re - tool for loading the jan release of cobol programs
% for the update subsystem
% 08/20/97

function go-update-jan () =
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/15jan97/lygdrwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/15jan97/lygctwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/15jan97/lygucwqs.u"));
NIL
```

```
!! in-package("RCBU")
!! in-grammar('user)

% load-crqs-apr.re - tool for loading apr release of cobol programs
% for the CRQS subsystem
% 08/20/97

function go-crqs-apr () =
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/crqs/baseline/30apr97/lbamnwqs.u"));
NIL
```

```
!! in-package("RCBU")
!! in-grammar('user)

% load-maca-apr.re - tool for loading the apr release cobol programs
% for the maca subsystem
% 08/20/97

function go-maca-apr () =
  seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/30apr97/lygadwqs.u"));
  seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/30apr97/lygbtwqs.u"));
  seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/30apr97/lygedwqs.u"));
  seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/30apr97/lygotwqs.u"));
  seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/maca/baseline/30apr97/lygrnwqs.u"));
NIL
```

```
!! in-package("RCBU")
!! in-grammar('user)

% load-trais-apr.re - tool for loading the apr release of cobol programs
% for the trais subsystem
% 08/20/97

function go-trais-apr () =
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/30apr97/lnxm1wqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/30apr97/lnxmpwqs.u"));
 seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/trais/baseline/30apr97/lnxrpwqs.u"));
 NIL
```

```
!! in-package("RCBU")
!! in-grammar('user)

% load-update-apr.re - tool for loading apr release of cobol programs
% for the update subsystem
% 08/20/97

function go-update-apr () =
  seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/30apr97/lygalwqs.u"));
  seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/30apr97/lygctwqs.u"));
  seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/30apr97/lygucwqs.u"));
  seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/30apr97/lygmpwqs.u"));
  seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/30apr97/lygprwqs.u"));
  seq-file <- append(seq-file, parse-cobol-file("~rboone/HOST/update/baseline/30apr97/lygpmwqs.u"));
NIL
```

## *Appendix B*

Baseline Source Code Measures

Table of Contents

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXA0WQ | 000_MAIN_DRIVER | 13 | | | | 7 | | | | 2 | | | | 31213 | | | |
| NXA0WQ | 000_STOP_RUN | 4 | | | | 1 | | | | 1 | | | | 64 | | | |
| NXA0WQ | 100_HSKP | 71 | | | | 15 | | | | 2 | | | | 18684999 | | | |
| NXA0WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA0WQ | 110_LOAD_DEST_TABLE | 16 | | | | 7 | | | | 2 | | | | 63504 | | | |
| NXA0WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA0WQ | 200_SEND_RCDS | 25 | | | | 12 | | | | 2 | | | | 562500 | | | |
| NXA0WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA0WQ | 700_RBTCHO_CALL | 8 | | | | 4 | | | | 2 | | | | 8192 | | | |
| NXA0WQ | 700_RBTCHO_CALL_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA0WQ | 700_SEND_HDRS | 64 | | | | 24 | | | | 3 | | | | 20070400 | | | |
| NXA0WQ | 700_SEND_HDRS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA0WQ | 710_FORMAT_RIS_AREA | 7 | | | | 3 | | | | 2 | | | | 4032 | | | |
| NXA0WQ | 710_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA0WQ | 720_SEND_PLA_RECS | 17 | | | | 6 | | | | 2 | | | | 108800 | | | |
| NXA0WQ | 720_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA0WQ | 700_READ_INPUT_FILE | 6 | | | | 5 | | | | 2 | | | | 216 | | | |
| NXA0WQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA0WQ | 700_CHECK_RBTCHO_RZLT | 8 | | | | 3 | | | | 2 | | | | 10368 | | | |
| NXA0WQ | 700_CHECK_RBTCHO_RZLT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA0WQ | 700_RBTCHO_LOOP | 5 | | | | 1 | | | | 1 | | | | 5445 | | | |
| NXA0WQ | 700_RBTCHO_LOOP_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA0WQ | 700_ERROR_DUMP | 10 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXA0WQ | 700_ERROR_DUMP_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA1WQ | 000_MAIN_DRIVER | 11 | | | | 6 | | | | 2 | | | | 14256 | | | |
| NXA1WQ | 000_STOP_RUN | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA1WQ | 100_HSKP | 74 | | | | 7 | | | | 2 | | | | 39976650 | | | |
| NXA1WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA1WQ | 200_SEND_RCDS | 26 | | | | 6 | | | | 3 | | | | 473850 | | | |
| NXA1WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA1WQ | 700_STARS_CALL | 9 | | | | 5 | | | | 2 | | | | 32400 | | | |
| NXA1WQ | 700_STARS_CALL_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA1WQ | 700_HDR_CALL | 42 | | | | 9 | | | | 2 | | | | 2183328 | | | |
| NXA1WQ | 700_HDR_CALL_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA1WQ | 700_READ_INPUT_FILE | 6 | | | | 4 | | | | 2 | | | | 600 | | | |
| NXA1WQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA1WQ | 700_CHECK_RBTCHO_RZLT | 7 | | | | 2 | | | | 2 | | | | 10647 | | | |
| NXA1WQ | 700_CHECK_RBTCHO_RZLT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA1WQ | 700_RBTCHO_LOOP | 5 | | | | 1 | | | | 1 | | | | 6480 | | | |
| NXA1WQ | 700_RBTCHO_LOOP_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA1WQ | 700_ERROR_DUMP | 9 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXA1WQ | 700_ERROR_DUMP_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXADWQ | 000_DRIVER | 4 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXADWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXADWQ | 100_HSKP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXADWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXADWQ | 200_READ_DRIVER | 5 | | | | 3 | | | | 2 | | | | 180 | | | |
| NXADWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXADWQ | 210_FIRST_READ | 6 | | | | 3 | | | | 2 | | | | 486 | | | |
| NXADWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXADWQ | 220_READ_REST | 9 | | | | 7 | | | | 2 | | | | 32400 | | | |
| NXADWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXADWQ | 230_PAGE_BREAK | 6 | | | | 5 | | | | 1 | | | | 1944 | | | |
| NXADWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXADWQ | 211_MOVE_INPUT | 6 | | | | 1 | | | | 1 | | | | 7350 | | | |
| NXADWQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXADWQ | 212_WRITE_REC | 8 | | | | 3 | | | | 2 | | | | 2048 | | | |
| NXADWQ | 212_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXADWQ | 300_WRAPUP | 4 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXADWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 000_B0_DRIVER | 25 | | | | 11 | | | | 3 | | | | 342225 | | | |
| NXB0WQ | 000_CONTINUE | 3 | | | | 2 | | | | 2 | | | | 12 | | | |
| NXB0WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 100_HSKP | 97 | | | | 23 | | | | 2 | | | | 655720000 | | | |
| NXB0WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 110_LOAD_PGM_STA_TABLE | 22 | | | | 9 | | | | 3 | | | | 130438 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXB0WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 120_LOAD_VALID_PORT_TABLE | 5 | | | | 2 | | | | 1 | | | | 405 | | | |
| NXB0WQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 130_DAILY_ONHAND_FILE | 46 | | | | 35 | | | | 3 | | | | 2308096 | | | |
| NXB0WQ | 130_MERGE | 31 | | | | 14 | | | | 3 | | | | 1697436 | | | |
| NXB0WQ | 130_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 131_READ_OH_FILE | 9 | | | | 4 | | | | 2 | | | | 20736 | | | |
| NXB0WQ | 131_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 140_DAILY_MOVEMENT_FILE | 30 | | | | 16 | | | | 3 | | | | 243000 | | | |
| NXB0WQ | 140_MERGE | 89 | | | | 41 | | | | 3 | | | | 88822089 | | | |
| NXB0WQ | 140_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 141_READ_MV_FILE | 10 | | | | 4 | | | | 2 | | | | 51840 | | | |
| NXB0WQ | 141_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 142_READ_NEW_GEN_FILE | 8 | | | | 4 | | | | 2 | | | | 512 | | | |
| NXB0WQ | 142_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 143_READ_WRITE_GEN | 12 | | | | 6 | | | | 2 | | | | 4800 | | | |
| NXB0WQ | 143_READ_WRITE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 150_LEAP_YEAR_CHECK | 16 | | | | 6 | | | | 3 | | | | 5184 | | | |
| NXB0WQ | 150_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 200_PROCESS | 21 | | | | 3 | | | | 2 | | | | 97104 | | | |
| NXB0WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 210_COMPARISONS | 13 | | | | 17 | | | | 2 | | | | 7488 | | | |
| NXB0WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 211_CONUS_OUTBOUND | 11 | | | | 8 | | | | 4 | | | | 8624 | | | |
| NXB0WQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 212_CONUS_INBOUND | 11 | | | | 8 | | | | 4 | | | | 8624 | | | |
| NXB0WQ | 212_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 213_OFFSHORE_TO_OFFSHORE | 11 | | | | 8 | | | | 4 | | | | 8624 | | | |
| NXB0WQ | 213_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 300_DELETE_OLD_TCN_RECS | 19 | | | | 7 | | | | 2 | | | | 112651 | | | |
| NXB0WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 310_READ_TO_TEMP | 8 | | | | 4 | | | | 2 | | | | 2048 | | | |
| NXB0WQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 400_DELE_AFTER_3_DAYS | 7 | | | | 4 | | | | 2 | | | | 1008 | | | |
| NXB0WQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 410_READ_TO_TEMP | 7 | | | | 4 | | | | 2 | | | | 1008 | | | |
| NXB0WQ | 410_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 420_READ_WRITE_GEN | 10 | | | | 4 | | | | 2 | | | | 5760 | | | |
| NXB0WQ | 420_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 500_WRAPUP | 51 | | | | 4 | | | | 2 | | | | 3316275 | | | |
| NXB0WQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 700_READ | 9 | | | | 4 | | | | 2 | | | | 2304 | | | |
| NXB0WQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 700_WRITE_DETAIL | 31 | | | | 4 | | | | 2 | | | | 8447004 | | | |
| NXB0WQ | 700_WRITE_DETAIL_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 700_WRITE_HEADINGS | 9 | | | | 9 | | | | 2 | | | | 729 | | | |
| NXB0WQ | 700_WRITE_HEADINGS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 700_WRITE_TRAILER | 4 | | | | 4 | | | | 2 | | | | 100 | | | |
| NXB0WQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 700_ERROR_RTN | 4 | | | | 1 | | | | 1 | | | | 400 | | | |
| NXB0WQ | 700_ERROR_RTN_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 700_SEARCH_VALID_PORT_TBL | 5 | | | | 5 | | | | 2 | | | | 0 | | | |
| NXB0WQ | 700_SEARCH_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 700_READ_TEMP_DELETE | 4 | | | | 3 | | | | 2 | | | | 100 | | | |
| NXB0WQ | 700_READ_TEMP_DELETE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXB0WQ | 700_READ_WRITE_GEN | 38 | | | | 21 | | | | 2 | | | | 493848 | | | |
| NXB0WQ | 700_READ_WRITE_GEN_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 000_BA_DRIVER | 8 | | | | 1 | | | | 1 | | | | 7200 | | | |
| NXBAWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 100_HSKP | 18 | | | | 3 | | | | 1 | | | | 3484800 | | | |
| NXBAWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 110_LOAD_VALID_PORT_TABLE | 6 | | | | 2 | | | | 1 | | | | 1350 | | | |
| NXBAWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 120_LOAD_MAI_TABLE | 6 | | | | 2 | | | | 1 | | | | 1350 | | | |
| NXBAWQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 200_PROCESS_OH_FILE | 3 | | | | 3 | | | | 1 | | | | 108 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXBAWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 210_L4_AKF_PROCESS | 13 | | | | 3 | | | | 2 | | | | 16848 | | | |
| NXBAWQ | 210_CONTINUE | 28 | | | | 8 | | | | 2 | | | | 5057500 | | | |
| NXBAWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 220_L5_AKF_PROCESS | 32 | | | | 9 | | | | 2 | | | | 4078368 | | | |
| NXBAWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 300_WRITE_CONTROL | 11 | | | | 6 | | | | 1 | | | | 57024 | | | |
| NXBAWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 310_WRITE_PART_II | 49 | | | | 14 | | | | 3 | | | | 8850625 | | | |
| NXBAWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 320_WRITE_PART_V | 47 | | | | 11 | | | | 3 | | | | 8133632 | | | |
| NXBAWQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 400_WRAPUP | 4 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 700_SEARCH_STN_WT_TABLE | 5 | | | | 4 | | | | 1 | | | | 1620 | | | |
| NXBAWQ | 700_SRCH_STN_WT_TABLE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 700_SEARCH_VALID_PORT_TBL | 7 | | | | 5 | | | | 2 | | | | 343 | | | |
| NXBAWQ | 700_SEARCH_VALID_PORT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 700_ADD_TO_MAI_TOTALS_B2 | 29 | | | | 24 | | | | 4 | | | | 234900 | | | |
| NXBAWQ | 700_ADD_TO_MAI_TOTALS_B2_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 700_ADD_TO_MAI_TOTALS_B5 | 29 | | | | 34 | | | | 4 | | | | 319725 | | | |
| NXBAWQ | 700_ADD_TO_MAI_TOTALS_B5_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 700_ADD_TO_REPORT_TOTALS | 17 | | | | 31 | | | | 2 | | | | 39168 | | | |
| NXBAWQ | 700_ADD_TO_REPORT_TOTALS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 700_FORMAT_MAI_PART | 9 | | | | 3 | | | | 1 | | | | 39204 | | | |
| NXBAWQ | 700_FORMAT_MAI_PART_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 700_FORMAT_MAI_CONUS_TOTALS | 7 | | | | 1 | | | | 1 | | | | 6300 | | | |
| NXBAWQ | 700_FORMAT_MAI_CONUS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 700_B2FORM_MAI_OFFSHORE_TOTALS | 7 | | | | 1 | | | | 1 | | | | 6300 | | | |
| NXBAWQ | 700_B2FORM_MAI_OFFSHORE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 700_FORMAT_MAI_OFFSHORE_TOTALS | 7 | | | | 1 | | | | 1 | | | | 6300 | | | |
| NXBAWQ | 700_FORMAT_MAI_OFFSHORE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 700_FORMAT_21ST_AF_TOTALS | 6 | | | | 1 | | | | 1 | | | | 5400 | | | |
| NXBAWQ | 700_FORMAT_21ST_AF_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 700_FORMAT_22ND_AF_TOTALS | 6 | | | | 1 | | | | 1 | | | | 5400 | | | |
| NXBAWQ | 700_FORMAT_22ND_AF_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 700_FORMAT_GRAND_TOTALS | 6 | | | | 1 | | | | 1 | | | | 15000 | | | |
| NXBAWQ | 700_FORMAT_GRAND_TOTL_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 700_TRAILER_PART_II | 7 | | | | 2 | | | | 1 | | | | 9072 | | | |
| NXBAWQ | 700_TRAILER_PART_II_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 700_TRAILER_PART_V | 7 | | | | 2 | | | | 1 | | | | 9072 | | | |
| NXBAWQ | 700_TRAILER_PART_V_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 700_HEADERS_PART_II | 10 | | | | 6 | | | | 1 | | | | 7840 | | | |
| NXBAWQ | 700_HEADERS_PART_II_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBAWQ | 700_HEADERS_PART_V | 10 | | | | 6 | | | | 1 | | | | 7840 | | | |
| NXBAWQ | 700_HEADERS_PART_V_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 000_BB_DRIVER | 9 | | | | 5 | | | | 1 | | | | 44100 | | | |
| NXBBWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 100_HSKP | 33 | | | | 6 | | | | 3 | | | | 21651300 | | | |
| NXBBWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 110_LOAD_B0_GENERATION_TABLE | 12 | | | | 8 | | | | 2 | | | | 15552 | | | |
| NXBBWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 120_LOAD_VALID_PORT_TABLE | 6 | | | | 2 | | | | 1 | | | | 1350 | | | |
| NXBBWQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 200_PREPARE_MVMT | 28 | | | | 16 | | | | 3 | | | | 1612800 | | | |
| NXBBWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 300_MERGE | 32 | | | | 5 | | | | 2 | | | | 663552 | | | |
| NXBBWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 310_READ_MV_SUMMARY_FILE | 4 | | | | 2 | | | | 2 | | | | 100 | | | |
| NXBBWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 320_READ_OH_SUMMARY_FILE | 4 | | | | 2 | | | | 2 | | | | 100 | | | |
| NXBBWQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 400_WRITE_CONTROL | 8 | | | | 8 | | | | 1 | | | | 3200 | | | |
| NXBBWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 410_WRITE_PART_I_CONUS_OUT | 42 | | | | 10 | | | | 2 | | | | 3780000 | | | |
| NXBBWQ | 410_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXBBWQ | 420_WRITE_PART_III_CONUS_IN | 53 | | | | 13 | | | | 2 | | | | 8607677 | | | |
| NXBBWQ | 420_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 430_WRITE_PART_IV_OFF_SHORE | 53 | | | | 11 | | | | 2 | | | | 8607677 | | | |
| NXBBWQ | 430_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 500_PREPARE_ON_HAND | 25 | | | | 12 | | | | 2 | | | | 950625 | | | |
| NXBBWQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 600_WRAPUP | 4 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 600_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_UPDATE_TOTLS_FROM_FILE_FS | 10 | | | | 1 | | | | 1 | | | | 29160 | | | |
| NXBBWQ | 700_UPDATE_TOTLS_FROM_FS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_ADD_TO_GENERATION_NAF_TOT | 20 | | | | 6 | | | | 2 | | | | 180500 | | | |
| NXBBWQ | 700_ADD_TO_GEN_NAF_TOT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_ADD_TO_GENERATION_RPT_TOT | 20 | | | | 6 | | | | 2 | | | | 200000 | | | |
| NXBBWQ | 700_ADD_TO_GEN_RPT_TOT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_FORMAT_DETAIL_LINE | 25 | | | | 6 | | | | 2 | | | | 1322500 | | | |
| NXBBWQ | 700_FORMAT_DETAIL_LINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_FORMAT_NAF_TOTAL_LINE | 16 | | | | 3 | | | | 2 | | | | 278784 | | | |
| NXBBWQ | 700_FORMAT_NAF_TOTAL_LINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_FORMAT_REPORT_TOTAL_LINE | 13 | | | | 2 | | | | 2 | | | | 236925 | | | |
| NXBBWQ | 700_FORMAT_RPT_TOTAL_LINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_SEARCH_VALID_PORT_TBL | 4 | | | | 4 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_SEARCH_VALID_PORT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_READ_MVMT | 24 | | | | 10 | | | | 2 | | | | 777600 | | | |
| NXBBWQ | 700_READ_MVMT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_READ_ONHD | 17 | | | | 7 | | | | 2 | | | | 137700 | | | |
| NXBBWQ | 700_READ_ONHD_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_PRI_4_INFO_FROM_LB | 22 | | | | 7 | | | | 3 | | | | 514998 | | | |
| NXBBWQ | 700_PRI_4_INFO_FROM_LB_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_READ_PRI_4 | 2 | | | | 2 | | | | 1 | | | | 8 | | | |
| NXBBWQ | 700_READ_PRI_4_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_READ_CONUS_PRI_4 | 2 | | | | 2 | | | | 1 | | | | 8 | | | |
| NXBBWQ | 700_READ_CONUS_PRI_4_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_READ_INTRA_PRI_4 | 2 | | | | 2 | | | | 1 | | | | 8 | | | |
| NXBBWQ | 700_READ_INTRA_PRI_4_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_TRAILER_PART_I | 7 | | | | 2 | | | | 1 | | | | 6300 | | | |
| NXBBWQ | 700_TRAILER_PART_I_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_TRAILER_PART_III | 7 | | | | 2 | | | | 1 | | | | 6300 | | | |
| NXBBWQ | 700_TRAILER_PART_III_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_TRAILER_PART_IV | 7 | | | | 2 | | | | 1 | | | | 6300 | | | |
| NXBBWQ | 700_TRAILER_PART_IV_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_HEADERS_PART_I | 10 | | | | 7 | | | | 1 | | | | 4410 | | | |
| NXBBWQ | 700_HEADERS_PART_I_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_HEADERS_PART_III | 10 | | | | 7 | | | | 1 | | | | 4410 | | | |
| NXBBWQ | 700_HEADERS_PART_III_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBBWQ | 700_HEADERS_PART_IV | 10 | | | | 7 | | | | 1 | | | | 4410 | | | |
| NXBBWQ | 700_HEADERS_PART_IV_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBCWQ | 000_BC_DRIVER | 7 | | | | 2 | | | | 1 | | | | 8575 | | | |
| NXBCWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBCWQ | 100_HSKP | 20 | | | | 3 | | | | 1 | | | | 3528000 | | | |
| NXBCWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBCWQ | 110_LOAD_VALID_PORT_TABLE | 6 | | | | 2 | | | | 1 | | | | 1350 | | | |
| NXBCWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBCWQ | 200_L5_SELECT_PROC | 39 | | | | 16 | | | | 2 | | | | 3510000 | | | |
| NXBCWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBCWQ | 300_WRITE_CONTROL | 7 | | | | 6 | | | | 1 | | | | 1575 | | | |
| NXBCWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBCWQ | 310_WRITE_SECTION_A | 18 | | | | 8 | | | | 2 | | | | 217800 | | | |
| NXBCWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBCWQ | 311_FORMAT_AND_WRITE_SECT_A | 16 | | | | 3 | | | | 2 | | | | 104976 | | | |
| NXBCWQ | 311_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBCWQ | 320_WRITE_SECTION_B | 22 | | | | 9 | | | | 2 | | | | 521752 | | | |
| NXBCWQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBCWQ | 321_FORMAT_AND_WRITE_SECT_B | 16 | | | | 3 | | | | 2 | | | | 104976 | | | |
| NXBCWQ | 321_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBCWQ | 400_WRAPUP | 3 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBCWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXBCWQ | 700_SEARCH_VALID_PORT_TBL | 4 | | | | 4 | | | | 1 | | | | 0 | | | |
| NXBCWQ | 700_SEARCH_VALID_PORT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBCWQ | 700_ADD_TO_DETAIL_CNTRS | 6 | | | | 2 | | | | 2 | | | | 1536 | | | |
| NXBCWQ | 700_ADD_TO_DETAIL_CNTRS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBCWQ | 700_TRAILER | 6 | | | | 2 | | | | 1 | | | | 9600 | | | |
| NXBCWQ | 700_TRAILER_PART_VI_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBCWQ | 700_HEADERS | 11 | | | | 7 | | | | 2 | | | | 5324 | | | |
| NXBCWQ | 700_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBXWQ | 000_DRIVER | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBXWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBXWQ | 100_SPAWN_B0 | 9 | | | | 1 | | | | 1 | | | | 18225 | | | |
| NXBXWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBXWQ | 110_CHECK_R_SPAWN_ERR | 5 | | | | 4 | | | | 2 | | | | 20 | | | |
| NXBXWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXBXWQ | 111_RWAIT | 7 | | | | 2 | | | | 2 | | | | 9072 | | | |
| NXBXWQ | 111_RWAIT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI0WQ | 000_DRIVER | 7 | | | | 3 | | | | 2 | | | | 1008 | | | |
| NXI0WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI0WQ | 100_HSKP | 22 | | | | 6 | | | | 2 | | | | 1901592 | | | |
| NXI0WQ | 100_HSKP_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI0WQ | 110_LOAD_CHNL_TABLE | 6 | | | | 2 | | | | 1 | | | | 486 | | | |
| NXI0WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI0WQ | 120_SELECT_RECORDS | 10 | | | | 10 | | | | 2 | | | | 16000 | | | |
| NXI0WQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI0WQ | 130_SEARCH_CHNL_TBL | 6 | | | | 3 | | | | 1 | | | | 600 | | | |
| NXI0WQ | 130_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI0WQ | 200_BUILD_REPORT | 28 | | | | 6 | | | | 2 | | | | 1075648 | | | |
| NXI0WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI0WQ | 210_EACH_APOD | 11 | | | | 6 | | | | 2 | | | | 57024 | | | |
| NXI0WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI0WQ | 220_EACH_PRIORITY | 36 | | | | 30 | | | | 2 | | | | 3572100 | | | |
| NXI0WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI0WQ | 230_PROCESS_LINE | 12 | | | | 5 | | | | 2 | | | | 139968 | | | |
| NXI0WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI0WQ | 231_COMPUTE | 18 | | | | 4 | | | | 3 | | | | 71442 | | | |
| NXI0WQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI0WQ | 232_WRITE_LINE | 27 | | | | 6 | | | | 3 | | | | 3345408 | | | |
| NXI0WQ | 232_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI0WQ | 300_WRAP_UP | 24 | | | | 4 | | | | 2 | | | | 279936 | | | |
| NXI0WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI0WQ | 600_NEXT_PAGE | 4 | | | | 3 | | | | 2 | | | | 784 | | | |
| NXI0WQ | 600_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI0WQ | 700_WRITE_HEADERS | 9 | | | | 6 | | | | 2 | | | | 9216 | | | |
| NXI0WQ | 700_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| LNXI1WQ | 000_DRIVER | 3 | | | | 2 | | | | 1 | | | | 27 | | | |
| LNXI1WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| LNXI1WQ | 100_HSKP | 19 | | | | 5 | | | | 1 | | | | 1076236 | | | |
| LNXI1WQ | 100_HSKP_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| LNXI1WQ | 110_LOAD_CHNL_TABLE | 5 | | | | 2 | | | | 1 | | | | 405 | | | |
| LNXI1WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| LNXI1WQ | 120_SELECT_RECORDS | 16 | | | | 11 | | | | 2 | | | | 40000 | | | |
| LNXI1WQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| LNXI1WQ | 130_SEARCH_CHNL_TBL | 5 | | | | 3 | | | | 1 | | | | 500 | | | |
| LNXI1WQ | 130_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| LNXI1WQ | 200_BUILD_REPORT | 49 | | | | 11 | | | | 2 | | | | 12446784 | | | |
| LNXI1WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| LNXI1WQ | 210_EACH_CHNL | 12 | | | | 6 | | | | 2 | | | | 209088 | | | |
| LNXI1WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| LNXI1WQ | 220_EACH_MFST_CHNL | 16 | | | | 8 | | | | 2 | | | | 94864 | | | |
| LNXI1WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| LNXI1WQ | 230_COMPUTE | 6 | | | | 1 | | | | 1 | | | | 29400 | | | |
| LNXI1WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| LNXI1WQ | 231_ADD_TOTALS | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| LNXI1WQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| LNXI1WQ | 232_WRITE_LINE | 18 | | | | 3 | | | | 2 | | | | 1843200 | | | |
| LNXI1WQ | 232_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| LNXI1WQ | 233_PERCENTAGES | 8 | | | | 1 | | | | 1 | | | | 20000 | | | |
| LNXI1WQ | 233_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| LNXI1WQ | 234_WRITE_PRCTS | 10 | | | | 4 | | | | 1 | | | | 59290 | | | |
| LNXI1WQ | 234_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| LNXI1WQ | 300_WRAP_UP | 32 | | | | 5 | | | | 1 | | | | 1952288 | | | |
| LNXI1WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| LNXI1WQ | 600_NEXT_PAGE | 2 | | | | 2 | | | | 1 | | | | 200 | | | |
| LNXI1WQ | 600_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| LNXI1WQ | 700_WRITE_HEADERS | 7 | | | | 5 | | | | 1 | | | | 5488 | | | |
| LNXI1WQ | 700_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 000_I2_DRIVER | 41 | | | | 9 | | | | 2 | | | | 410000 | | | |
| NXI2WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 100_HSKP | 24 | | | | 3 | | | | 1 | | | | 3538944 | | | |
| NXI2WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 110_SELECT_RECORDS | 22 | | | | 9 | | | | 2 | | | | 301158 | | | |
| NXI2WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 200_OUTBOUND_SHIPMENTS | 26 | | | | 20 | | | | 2 | | | | 842400 | | | |
| NXI2WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 210_APOE_LOOP | 17 | | | | 10 | | | | 2 | | | | 78608 | | | |
| NXI2WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 220_COMPUTE_APOE | 19 | | | | 2 | | | | 1 | | | | 1076236 | | | |
| NXI2WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 230_TOTAL_REC | 18 | | | | 3 | | | | 1 | | | | 684450 | | | |
| NXI2WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 300_SHIPMENTS_TONS | 25 | | | | 29 | | | | 2 | | | | 435600 | | | |
| NXI2WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 310_TONS_LOOP | 3 | | | | 1 | | | | 1 | | | | 432 | | | |
| NXI2WQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 320_COMPUTE_TONS | 7 | | | | 2 | | | | 1 | | | | 9072 | | | |
| NXI2WQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 330_TOTAL_TON | 8 | | | | 3 | | | | 1 | | | | 7200 | | | |
| NXI2WQ | 330_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 400_RETRO_OUTBND_SHPMTS | 22 | | | | 20 | | | | 2 | | | | 508288 | | | |
| NXI2WQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 410_RETRO_APOE_LOOP | 17 | | | | 10 | | | | 2 | | | | 78608 | | | |
| NXI2WQ | 410_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 420_COMPUTE_RETRO_APOE | 18 | | | | 2 | | | | 1 | | | | 903168 | | | |
| NXI2WQ | 420_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 430_RETRO_TOTAL_REC | 18 | | | | 3 | | | | 1 | | | | 684450 | | | |
| NXI2WQ | 430_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 500_RETRO_SHPMTS_TONS | 26 | | | | 29 | | | | 2 | | | | 453024 | | | |
| NXI2WQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 510_RETRO_TONS_LOOP | 3 | | | | 1 | | | | 1 | | | | 432 | | | |
| NXI2WQ | 510_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 520_COMPUTE_RETRO_TONS | 7 | | | | 2 | | | | 1 | | | | 9072 | | | |
| NXI2WQ | 520_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 530_TOTAL_RETRO_TONS | 8 | | | | 3 | | | | 1 | | | | 7200 | | | |
| NXI2WQ | 530_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 600_WRAPUP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 600_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 700_WRITE_HEADERS | 9 | | | | 8 | | | | 1 | | | | 2304 | | | |
| NXI2WQ | 700_WRITE_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 700_TONS_HEADERS | 10 | | | | 9 | | | | 1 | | | | 3240 | | | |
| NXI2WQ | 700_TONS_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 700_WRITE_RETRO_HDRS | 9 | | | | 8 | | | | 1 | | | | 2304 | | | |
| NXI2WQ | 700_WRITE_RETRO_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 700_RETRO_TONS_HDRS | 10 | | | | 9 | | | | 1 | | | | 3240 | | | |
| NXI2WQ | 700_RETRO_TONS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 700_INIT_REPORT | 2 | | | | 1 | | | | 1 | | | | 32 | | | |
| NXI2WQ | 700_INIT_REPORT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 700_INIT_RETRO | 2 | | | | 1 | | | | 1 | | | | 18 | | | |
| NXI2WQ | 700_INIT_RETRO_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 700_INIT_TONS | 2 | | | | 1 | | | | 1 | | | | 18 | | | |
| NXI2WQ | 700_INIT_TONS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI2WQ | 700_INIT_RETRO_TONS | 2 | | | | 1 | | | | 1 | | | | 18 | | | |
| NXI2WQ | 700_INIT_RETRO_TONS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXI3WQ | 000_I3_DRIVER | 4 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI3WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI3WQ | 100_HSKP | 11 | | | | 3 | | | | 1 | | | | 53900 | | | |
| NXI3WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI3WQ | 110_SELECT_RECORDS | 17 | | | | 21 | | | | 2 | | | | 4352 | | | |
| NXI3WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI3WQ | 200_HAZCGO_PROCESS | 7 | | | | 3 | | | | 1 | | | | 4032 | | | |
| NXI3WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI3WQ | 210_HAZCGO_COUNT | 11 | | | | 4 | | | | 1 | | | | 174636 | | | |
| NXI3WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI3WQ | 220_COMPUTE_HAZCGO | 7 | | | | 2 | | | | 1 | | | | 14175 | | | |
| NXI3WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI3WQ | 230_HAZCGO_TOTAL | 11 | | | | 3 | | | | 1 | | | | 43659 | | | |
| NXI3WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI3WQ | 300_WRAPUP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI3WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI3WQ | 700_WRITE_HEADERS | 7 | | | | 6 | | | | 1 | | | | 1008 | | | |
| NXI3WQ | 700_WRITE_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 000_I4_DRIVER | 16 | | | | 6 | | | | 2 | | | | 7744 | | | |
| NXI4WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 100_HSKP | 35 | | | | 6 | | | | 2 | | | | 54512640 | | | |
| NXI4WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 105_LOAD_I4_STATION_TABLE | 7 | | | | 3 | | | | 2 | | | | 1008 | | | |
| NXI4WQ | 105_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 110_SELECT_RECORDS | 18 | | | | 8 | | | | 2 | | | | 31752 | | | |
| NXI4WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 200_PROCESS_RPT_1 | 21 | | | | 11 | | | | 3 | | | | 162624 | | | |
| NXI4WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 300_PROCESS_RPT_2 | 21 | | | | 11 | | | | 3 | | | | 162624 | | | |
| NXI4WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 400_PROCESS_RPT_3 | 21 | | | | 12 | | | | 3 | | | | 162624 | | | |
| NXI4WQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 500_PROCESS_RPT_4 | 21 | | | | 12 | | | | 3 | | | | 162624 | | | |
| NXI4WQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 600_WRAPUP | 3 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXI4WQ | 600_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 700_COMPUTE | 6 | | | | 5 | | | | 2 | | | | 153600 | | | |
| NXI4WQ | 700_COMPUTE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 710_ADD | 41 | | | | 31 | | | | 3 | | | | 22694976 | | | |
| NXI4WQ | 710_CONT | 3 | | | | 2 | | | | 2 | | | | 3 | | | |
| NXI4WQ | 710_ADD_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 720_COMPUTE_SVC | 142 | | | | 42 | | | | 3 | | | | 801643392 | | | |
| NXI4WQ | 720_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 730_PORT_TOTAL | 118 | | | | 34 | | | | 3 | | | | 1097695000 | | | |
| NXI4WQ | 730_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 740_GRAND_TOTAL | 134 | | | | 34 | | | | 3 | | | | 2013132384 | | | |
| NXI4WQ | 740_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 700_PREP | 18 | | | | 6 | | | | 2 | | | | 368082 | | | |
| NXI4WQ | 700_PREP_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 700_READ | 2 | | | | 2 | | | | 1 | | | | 72 | | | |
| NXI4WQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 700_CE_HEADERS | 17 | | | | 6 | | | | 2 | | | | 71825 | | | |
| NXI4WQ | 700_CE_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 700_CD_HEADERS | 17 | | | | 6 | | | | 2 | | | | 71825 | | | |
| NXI4WQ | 700_CD_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 700_OE_HEADERS | 17 | | | | 6 | | | | 2 | | | | 71825 | | | |
| NXI4WQ | 700_OE_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 700_OD_HEADERS | 17 | | | | 6 | | | | 2 | | | | 71825 | | | |
| NXI4WQ | 700_OD_HEADERS_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| NXI4WQ | 700_TRAILERS | 12 | | | | 6 | | | | 2 | | | | 248832 | | | |
| NXI4WQ | 700_TRAILERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI4WQ | 700_SEARCH_STATION | 6 | | | | 4 | | | | 2 | | | | 1944 | | | |
| NXI4WQ | 700_SEARCH_STATION_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI5WQ | 000_I5_DRIVER | 6 | | | | 3 | | | | 2 | | | | 384 | | | |
| NXI5WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI5WQ | 100_HSKP | 29 | | | | 5 | | | | 2 | | | | 9588125 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXI5WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI5WQ | 110_SELECT_RECORDS | 15 | | | | 13 | | | | 2 | | | | 8640 | | | |
| NXI5WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI5WQ | 200_CONUS_OUTBOUND | 8 | | | | 5 | | | | 2 | | | | 5000 | | | |
| NXI5WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI5WQ | 210_MFST_STA | 6 | | | | 5 | | | | 2 | | | | 29400 | | | |
| NXI5WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI5WQ | 211_PRIORITY_SORT | 31 | | | | 14 | | | | 3 | | | | 12303900 | | | |
| NXI5WQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI5WQ | 212_COMPUTE_PRIORITIES | 109 | | | | 32 | | | | 3 | | | | 1609782741 | | | |
| NXI5WQ | 212_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI5WQ | 220_PORT_TOTAL | 99 | | | | 30 | | | | 2 | | | | 1546994691 | | | |
| NXI5WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI5WQ | 230_REPORT_TOTAL | 97 | | | | 28 | | | | 2 | | | | 1250842257 | | | |
| NXI5WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI5WQ | 300_WRAPUP | 5 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXI5WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI5WQ | 700_HEADERS | 9 | | | | 6 | | | | 2 | | | | 11664 | | | |
| NXI5WQ | 700_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI5WQ | 700_TOTAL_HDRS | 11 | | | | 3 | | | | 2 | | | | 8624 | | | |
| NXI5WQ | 700_TOTAL_HDRS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI5WQ | 700_NAVY_HEADERS | 9 | | | | 6 | | | | 2 | | | | 11664 | | | |
| NXI5WQ | 700_NAVY_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI5WQ | 700_TRAILERS | 8 | | | | 3 | | | | 2 | | | | 41472 | | | |
| NXI5WQ | 700_TRAILERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI6WQ | 000_I6_DRIVER | 4 | | | | 2 | | | | 1 | | | | 36 | | | |
| NXI6WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI6WQ | 100_HSKP | 20 | | | | 4 | | | | 1 | | | | 2178000 | | | |
| NXI6WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI6WQ | 110_SELECT_RECORDS | 26 | | | | 30 | | | | 2 | | | | 314600 | | | |
| NXI6WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI6WQ | 200_SHIP_IT | 8 | | | | 4 | | | | 2 | | | | 8192 | | | |
| NXI6WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI6WQ | 210_APOE_LOOP | 20 | | | | 7 | | | | 2 | | | | 619520 | | | |
| NXI6WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI6WQ | 211_APOD_LOOP | 7 | | | | 2 | | | | 1 | | | | 14175 | | | |
| NXI6WQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI6WQ | 220_APOE_TOTAL | 13 | | | | 3 | | | | 2 | | | | 56628 | | | |
| NXI6WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI6WQ | 230_REPORT_TOTAL | 14 | | | | 3 | | | | 2 | | | | 50400 | | | |
| NXI6WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI6WQ | 300_WRAPUP | 12 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI6WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI6WQ | 700_HEADERS | 6 | | | | 6 | | | | 1 | | | | 1944 | | | |
| NXI6WQ | 700_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI6WQ | 700_TRAILERS | 4 | | | | 2 | | | | 1 | | | | 2304 | | | |
| NXI6WQ | 700_TRAILERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 000_I7_DRIVER | 6 | | | | 3 | | | | 1 | | | | 150 | | | |
| NXI7WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 100_HSKP | 31 | | | | 3 | | | | 1 | | | | 10718064 | | | |
| NXI7WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 110_SELECT_RECORDS | 16 | | | | 12 | | | | 2 | | | | 16384 | | | |
| NXI7WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 200_MFSTSTA_NOT_APOE | 15 | | | | 4 | | | | 2 | | | | 45375 | | | |
| NXI7WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 210_SERVICES | 12 | | | | 8 | | | | 2 | | | | 28812 | | | |
| NXI7WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 211_SERVICE_SORT | 27 | | | | 6 | | | | 2 | | | | 3158028 | | | |
| NXI7WQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 212_COMPUTE_SVCS | 44 | | | | 4 | | | | 3 | | | | 21560000 | | | |
| NXI7WQ | 212_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 220_MFST_STA_TOTAL | 48 | | | | 2 | | | | 2 | | | | 75480768 | | | |
| NXI7WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 230_REPORT_TOTAL | 49 | | | | 2 | | | | 2 | | | | 81162081 | | | |
| NXI7WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 300_INTERIM | 16 | | | | 1 | | | | 1 | | | | 43264 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXI7WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 400_MFSTSTA_APOE | 13 | | | | 4 | | | | 2 | | | | 32500 | | | |
| NXI7WQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 410_SERVICES | 12 | | | | 8 | | | | 2 | | | | 28812 | | | |
| NXI7WQ | 410_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 411_SERVICE_SORT | 28 | | | | 7 | | | | 2 | | | | 3161088 | | | |
| NXI7WQ | 411_CONTINUE | 3 | | | | 1 | | | | 1 | | | | 108 | | | |
| NXI7WQ | 411_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 412_COMPUTE_SVCS | 44 | | | | 4 | | | | 3 | | | | 21560000 | | | |
| NXI7WQ | 412_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 420_APOE_TOTAL | 48 | | | | 2 | | | | 2 | | | | 75480768 | | | |
| NXI7WQ | 420_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 430_REPORT_TOTAL | 49 | | | | 2 | | | | 2 | | | | 81162081 | | | |
| NXI7WQ | 430_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 500_WRAPUP | 6 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 700_HEADERS | 9 | | | | 7 | | | | 1 | | | | 20736 | | | |
| NXI7WQ | 700_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 700_ORG_HEADERS | 8 | | | | 6 | | | | 1 | | | | 15488 | | | |
| NXI7WQ | 700_ORG_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 700_TRAILERS | 5 | | | | 2 | | | | 2 | | | | 18000 | | | |
| NXI7WQ | 700_TRAILERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI7WQ | 700_READ | 2 | | | | 2 | | | | 1 | | | | 50 | | | |
| NXI7WQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI8WQ | 000_I8_DRIVER | 4 | | | | 2 | | | | 1 | | | | 36 | | | |
| NXI8WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI8WQ | 100_HSKP | 26 | | | | 4 | | | | 1 | | | | 5033600 | | | |
| NXI8WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI8WQ | 110_SELECT_RECORDS | 15 | | | | 10 | | | | 2 | | | | 8640 | | | |
| NXI8WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI8WQ | 200_NTAC | 7 | | | | 4 | | | | 2 | | | | 7168 | | | |
| NXI8WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI8WQ | 210_APOE_LOOP | 4 | | | | 4 | | | | 1 | | | | 1296 | | | |
| NXI8WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI8WQ | 211_APOD_LOOP | 27 | | | | 8 | | | | 2 | | | | 2430000 | | | |
| NXI8WQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI8WQ | 211A_TAC_LOOP | 9 | | | | 2 | | | | 1 | | | | 97344 | | | |
| NXI8WQ | 211A_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI8WQ | 220_APOE_TOTAL | 16 | | | | 3 | | | | 2 | | | | 97344 | | | |
| NXI8WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI8WQ | 230_REPORT_TOTAL | 16 | | | | 3 | | | | 2 | | | | 112896 | | | |
| NXI8WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI8WQ | 300_WRAPUP | 4 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI8WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI8WQ | 700_HEADERS | 8 | | | | 5 | | | | 1 | | | | 10368 | | | |
| NXI8WQ | 700_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI8WQ | 700_TRAILERS | 5 | | | | 2 | | | | 1 | | | | 2880 | | | |
| NXI8WQ | 700_TRAILERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI9WQ | 000_I9_DRIVER | 3 | | | | 2 | | | | 1 | | | | 27 | | | |
| NXI9WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI9WQ | 100_HSKP | 19 | | | | 5 | | | | 1 | | | | 1710000 | | | |
| NXI9WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI9WQ | 110_LOAD_CHNL_TABLE | 5 | | | | 2 | | | | 1 | | | | 405 | | | |
| NXI9WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI9WQ | 120_SELECT_RECORDS | 24 | | | | 10 | | | | 2 | | | | 405600 | | | |
| NXI9WQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI9WQ | 130_SEARCH_APOE | 5 | | | | 3 | | | | 1 | | | | 500 | | | |
| NXI9WQ | 130_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI9WQ | 200_BUILD_REPORT | 13 | | | | 6 | | | | 2 | | | | 83200 | | | |
| NXI9WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI9WQ | 210_EACH_APOD | 10 | | | | 5 | | | | 2 | | | | 100000 | | | |
| NXI9WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI9WQ | 215_ADD_RECORDS | 30 | | | | 12 | | | | 2 | | | | 4704480 | | | |
| NXI9WQ | 215_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI9WQ | 220_ADD_PALLETS | 2 | | | | 2 | | | | 1 | | | | 8 | | | |
| NXI9WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXI9WQ | 231_COMPUTE | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI9WQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI9WQ | 232_WRITE_LINE | 45 | | | | 3 | | | | 2 | | | | 21611205 | | | |
| NXI9WQ | 232_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI9WQ | 300_WRAP_UP | 6 | | | | 2 | | | | 1 | | | | 3750 | | | |
| NXI9WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI9WQ | 600_NEXT_PAGE | 3 | | | | 2 | | | | 1 | | | | 972 | | | |
| NXI9WQ | 600_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXI9WQ | 700_WRITE_HEADERS | 9 | | | | 6 | | | | 1 | | | | 18225 | | | |
| NXI9WQ | 700_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIAWQ | 000_I9_DRIVER | 3 | | | | 2 | | | | 1 | | | | 27 | | | |
| NXIAWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIAWQ | 100_HSKP | 19 | | | | 5 | | | | 1 | | | | 1710000 | | | |
| NXIAWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIAWQ | 110_LOAD_CHNL_TABLE | 5 | | | | 2 | | | | 1 | | | | 405 | | | |
| NXIAWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIAWQ | 120_SELECT_RECORDS | 24 | | | | 10 | | | | 2 | | | | 405600 | | | |
| NXIAWQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIAWQ | 130_SEARCH_APOE | 5 | | | | 3 | | | | 1 | | | | 500 | | | |
| NXIAWQ | 130_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIAWQ | 200_BUILD_REPORT | 13 | | | | 6 | | | | 2 | | | | 83200 | | | |
| NXIAWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIAWQ | 210_EACH_APOD | 10 | | | | 5 | | | | 2 | | | | 100000 | | | |
| NXIAWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIAWQ | 215_ADD_RECORDS | 30 | | | | 12 | | | | 2 | | | | 4704480 | | | |
| NXIAWQ | 215_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIAWQ | 220_ADD_PALLETS | 2 | | | | 2 | | | | 1 | | | | 8 | | | |
| NXIAWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIAWQ | 231_COMPUTE | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIAWQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIAWQ | 232_WRITE_LINE | 45 | | | | 3 | | | | 2 | | | | 21611205 | | | |
| NXIAWQ | 232_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIAWQ | 300_WRAP_UP | 6 | | | | 2 | | | | 1 | | | | 3750 | | | |
| NXIAWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIAWQ | 600_NEXT_PAGE | 3 | | | | 2 | | | | 1 | | | | 972 | | | |
| NXIAWQ | 600_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIAWQ | 700_WRITE_HEADERS | 9 | | | | 6 | | | | 1 | | | | 18225 | | | |
| NXIAWQ | 700_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIBWQ | 000_I9_DRIVER | 3 | | | | 2 | | | | 1 | | | | 27 | | | |
| NXIBWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIBWQ | 100_HSKP | 20 | | | | 5 | | | | 1 | | | | 1984500 | | | |
| NXIBWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIBWQ | 110_LOAD_CHNL_TABLE | 5 | | | | 2 | | | | 1 | | | | 405 | | | |
| NXIBWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIBWQ | 120_SELECT_RECORDS | 25 | | | | 10 | | | | 2 | | | | 490000 | | | |
| NXIBWQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIBWQ | 130_SEARCH_APOE | 5 | | | | 3 | | | | 1 | | | | 500 | | | |
| NXIBWQ | 130_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIBWQ | 200_BUILD_REPORT | 13 | | | | 6 | | | | 2 | | | | 83200 | | | |
| NXIBWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIBWQ | 210_EACH_APOD | 10 | | | | 5 | | | | 2 | | | | 100000 | | | |
| NXIBWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIBWQ | 215_ADD_RECORDS | 30 | | | | 12 | | | | 2 | | | | 4704480 | | | |
| NXIBWQ | 215_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIBWQ | 220_ADD_PALLETS | 2 | | | | 2 | | | | 1 | | | | 8 | | | |
| NXIBWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIBWQ | 231_COMPUTE | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIBWQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIBWQ | 232_WRITE_LINE | 45 | | | | 3 | | | | 2 | | | | 21611205 | | | |
| NXIBWQ | 232_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIBWQ | 300_WRAP_UP | 6 | | | | 2 | | | | 1 | | | | 3750 | | | |
| NXIBWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIBWQ | 600_NEXT_PAGE | 3 | | | | 2 | | | | 1 | | | | 972 | | | |
| NXIBWQ | 600_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXIBWQ | 700_WRITE_HEADERS | 9 | | | | 6 | | | | 1 | | | | 18225 | | | |
| NXIBWQ | 700_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXICWQ | 000_IC_DRIVER | 6 | 6 | | | 3 | 3 | | | 2 | 2 | | | 384 | 384 | | |
| NXICWQ | 000_STOP_RUN | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 100_HSKP | 42 | 43 | | | 9 | 9 | | | 2 | 2 | | | 40336800 | 44299675 | | |
| NXICWQ | 100_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 110_SELECT_RECORDS | 5 | 5 | | | 5 | 5 | | | 2 | 2 | | | 1125 | 1125 | | |
| NXICWQ | 110_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 111_ONHD_RECORDS | 9 | 9 | | | 3 | 3 | | | 2 | 2 | | | 2304 | 2304 | | |
| NXICWQ | 111_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 112_OLD_MONTH_RECORDS | 15 | 15 | | | 9 | 9 | | | 2 | 2 | | | 11760 | 11760 | | |
| NXICWQ | 112_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 113_HISTORY_RECORDS | 11 | 11 | | | 8 | 8 | | | 2 | 2 | | | 3564 | 3564 | | |
| NXICWQ | 113_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 200_REPORT_BUILDER | 5 | 5 | | | 4 | 4 | | | 2 | 2 | | | 1125 | 1125 | | |
| NXICWQ | 200_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 210_NAF_LOOP | 5 | 5 | | | 5 | 5 | | | 2 | 2 | | | 2205 | 2205 | | |
| NXICWQ | 210_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 211_APOE_LOOP | 7 | 7 | | | 6 | 6 | | | 2 | 2 | | | 14175 | 14175 | | |
| NXICWQ | 211_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 212_APOE_TOTAL | 12 | 12 | | | 2 | 2 | | | 2 | 2 | | | 24300 | 24300 | | |
| NXICWQ | 212_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 2111_APOD_LOOP | 27 | 27 | | | 11 | 11 | | | 2 | 2 | | | 97200 | 97200 | | |
| NXICWQ | 2111_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 2112_ALPHA_DAY | 107 | 107 | | | 9 | 9 | | | 2 | 2 | | | 3019968 | 3019968 | | |
| NXICWQ | 2112_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 2113_APOD_TOTAL | 15 | 15 | | | 3 | 3 | | | 2 | 2 | | | 77760 | 77760 | | |
| NXICWQ | 2113_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 220_NAF_TOTAL | 13 | 13 | | | 2 | 2 | | | 2 | 2 | | | 20800 | 20800 | | |
| NXICWQ | 220_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 300_WRAPUP | 9 | 9 | | | 2 | 2 | | | 2 | 2 | | | 57600 | 57600 | | |
| NXICWQ | 300_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 700_HEADERS | 9 | 9 | | | 6 | 6 | | | 2 | 2 | | | 9216 | 9216 | | |
| NXICWQ | 700_HEADERS_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 700_DETERMINE_DAY_NUM | 29 | 29 | | | 9 | 9 | | | 2 | 2 | | | 678861 | 678861 | | |
| NXICWQ | 700_DAY_NUM_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 700_TRAILER | 5 | 5 | | | 3 | 3 | | | 2 | 2 | | | 720 | 720 | | |
| NXICWQ | 700_TRAILER_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 700_COMPUTE | 16 | 16 | | | 2 | 2 | | | 2 | 2 | | | 1254400 | 1254400 | | |
| NXICWQ | 700_COMPUTE_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 700_NEW_PAGE | 7 | 7 | | | 4 | 4 | | | 2 | 2 | | | 5488 | 5488 | | |
| NXICWQ | 700_NEW_PAGE_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXICWQ | 700_SELECT_AND_MOVE | 20 | 18 | | | 20 | 19 | | | 2 | 2 | | | 208080 | 145800 | | |
| NXICWQ | 700_SELECT_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 000_DRIVER | 6 | 6 | | | 3 | 3 | | | 2 | 2 | | | 384 | 384 | | |
| NXIDWQ | 000_STOP_RUN | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 100_HSKP | 31 | 31 | | | 4 | 4 | | | 2 | 2 | | | 21458944 | 21458944 | | |
| NXIDWQ | 100_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 110_SELECT | 23 | 19 | | | 6 | 6 | | | 2 | 2 | | | 1195632 | 109744 | | |
| NXIDWQ | 110_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 111_SELECT_ONHAND | 11 | 11 | | | 3 | 3 | | | 2 | 2 | | | 4400 | 4400 | | |
| NXIDWQ | 111_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 112_SELECT_HISTORY | 6 | 6 | | | 2 | 2 | | | 1 | 1 | | | 864 | 864 | | |
| NXIDWQ | 112_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 113_SELECT_OLD_PZ3 | 11 | 11 | | | 3 | 3 | | | 2 | 2 | | | 4400 | 4400 | | |
| NXIDWQ | 113_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 114_SELECT_PZ5_TAPE | 6 | 6 | | | 2 | 2 | | | 1 | 1 | | | 864 | 864 | | |
| NXIDWQ | 114_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 200_BUILDER | 5 | 5 | | | 4 | 4 | | | 2 | 2 | | | 1125 | 1125 | | |
| NXIDWQ | 200_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 210_NAF | 5 | 5 | | | 5 | 5 | | | 2 | 2 | | | 2205 | 2205 | | |
| NXIDWQ | 210_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 211_APOE | 5 | 5 | | | 6 | 6 | | | 2 | 2 | | | 3645 | 3645 | | |
| NXIDWQ | 211_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 2111_APOD | 24 | 24 | | | 9 | 9 | | | 2 | 2 | | | 375000 | 375000 | | |
| NXIDWQ | 2111_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 2112_APOD_TOTAL | 17 | 17 | | | 4 | 4 | | | 2 | 2 | | | 67473 | 67473 | | |
| NXIDWQ | 2112_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXIDWQ | 212_APOE_TOTAL | 10 | 10 | | | 2 | 2 | | | 2 | 2 | | | 9000 | 9000 | | |
| NXIDWQ | 212_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 220_NAF_TOTAL | 9 | 9 | | | 2 | 2 | | | 2 | 2 | | | 5625 | 5625 | | |
| NXIDWQ | 220_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 300_WRAP_UP | 14 | 14 | | | 2 | 2 | | | 2 | 2 | | | 243936 | 243936 | | |
| NXIDWQ | 300_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 710_COMPUTE | 45 | 45 | | | 14 | 14 | | | 2 | 2 | | | 4665780 | 4665780 | | |
| NXIDWQ | 710_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 711_WRITE_AMC | 10 | 10 | | | 5 | 5 | | | 2 | 2 | | | 12960 | 12960 | | |
| NXIDWQ | 711_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 7111_AMC_TRAILER | 7 | 7 | | | 3 | 3 | | | 2 | 2 | | | 16128 | 16128 | | |
| NXIDWQ | 7111_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 7112_AMC_HEADER | 6 | 6 | | | 5 | 5 | | | 2 | 2 | | | 1944 | 1944 | | |
| NXIDWQ | 7112_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 712_WRITE_ARMY | 8 | 8 | | | 4 | 4 | | | 2 | 2 | | | 10368 | 10368 | | |
| NXIDWQ | 712_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 7121_ARMY_TRAILER | 7 | 7 | | | 3 | 3 | | | 2 | 2 | | | 16128 | 16128 | | |
| NXIDWQ | 7121_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 7122_ARMY_HEADER | 6 | 6 | | | 5 | 5 | | | 2 | 2 | | | 1944 | 1944 | | |
| NXIDWQ | 7122_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 713_WRITE_NAVY | 8 | 8 | | | 4 | 4 | | | 2 | 2 | | | 10368 | 10368 | | |
| NXIDWQ | 713_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 7131_NAVY_TRAILER | 7 | 7 | | | 3 | 3 | | | 2 | 2 | | | 16128 | 16128 | | |
| NXIDWQ | 7131_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 7132_NAVY_HEADER | 6 | 6 | | | 5 | 5 | | | 2 | 2 | | | 1944 | 1944 | | |
| NXIDWQ | 7132_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 714_WRITE_USAF | 8 | 8 | | | 4 | 4 | | | 2 | 2 | | | 10368 | 10368 | | |
| NXIDWQ | 714_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 7141_USAF_TRAILER | 7 | 7 | | | 3 | 3 | | | 2 | 2 | | | 16128 | 16128 | | |
| NXIDWQ | 7141_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 7142_USAF_HEADER | 6 | 6 | | | 5 | 5 | | | 2 | 2 | | | 1944 | 1944 | | |
| NXIDWQ | 7142_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 715_WRITE_USMC | 8 | 8 | | | 4 | 4 | | | 2 | 2 | | | 10368 | 10368 | | |
| NXIDWQ | 715_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 7151_USMC_TRAILER | 7 | 7 | | | 3 | 3 | | | 2 | 2 | | | 16128 | 16128 | | |
| NXIDWQ | 7151_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 7152_USMC_HEADER | 6 | 6 | | | 5 | 5 | | | 2 | 2 | | | 1944 | 1944 | | |
| NXIDWQ | 7152_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 716_WRITE_DLA | 8 | 8 | | | 4 | 4 | | | 2 | 2 | | | 10368 | 10368 | | |
| NXIDWQ | 716_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 7161_DLA_TRAILER | 7 | 7 | | | 3 | 3 | | | 2 | 2 | | | 16128 | 16128 | | |
| NXIDWQ | 7161_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 7162_DLA_HEADER | 6 | 6 | | | 5 | 5 | | | 2 | 2 | | | 1944 | 1944 | | |
| NXIDWQ | 7162_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 717_WRITE_DECA | 8 | 8 | | | 4 | 4 | | | 2 | 2 | | | 10368 | 10368 | | |
| NXIDWQ | 717_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 7171_DECA_TRAILER | 7 | 7 | | | 3 | 3 | | | 2 | 2 | | | 16128 | 16128 | | |
| NXIDWQ | 7171_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 7172_DECA_HEADER | 6 | 6 | | | 5 | 5 | | | 2 | 2 | | | 1944 | 1944 | | |
| NXIDWQ | 7172_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 720_PROCESS | 32 | 32 | | | 19 | 18 | | | 2 | 2 | | | 4333568 | 3612672 | | |
| NXIDWQ | 720_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 730_HEADERS | 19 | 19 | | | 7 | 7 | | | 1 | 1 | | | 14896 | 14896 | | |
| NXIDWQ | 730_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 740_TRAILERS | 7 | 7 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXIDWQ | 740_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXL0WQ | 000_DRIVER | 11 | | | | 4 | | | | 2 | | | | 8624 | | | |
| NXL0WQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL0WQ | 100_RSPAWN_L2 | 15 | | | | 2 | | | | 2 | | | | 47040 | | | |
| NXL0WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL0WQ | 200_RSPAWN_S0 | 15 | | | | 2 | | | | 2 | | | | 47040 | | | |
| NXL0WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL0WQ | 400_RSPAWN_M0 | 18 | | | | 3 | | | | 2 | | | | 73728 | | | |
| NXL0WQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL0WQ | 700_RS_ABORT | 15 | | | | 6 | | | | 2 | | | | 2160 | | | |
| NXL0WQ | 700_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXL1WQ | 000_L1_KEY_BUILD_DRIVER | 24 | | | | 8 | | | | 2 | | | | 1280664 | | | |
| NXL1WQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 100_PROCESS_DATA_DRIVER | 100 | | | | 18 | | | | 3 | | | | 116208400 | | | |
| NXL1WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 105_LOAD_PGM_TABLES | 24 | | | | 12 | | | | 3 | | | | 470400 | | | |
| NXL1WQ | 105_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 110_PROCESS_MOVEMENT | 59 | | | | 21 | | | | 2 | | | | 30585600 | | | |
| NXL1WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 111_READ_MOVEMENT | 12 | | | | 4 | | | | 2 | | | | 15552 | | | |
| NXL1WQ | 111_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 112_SELECT_LG_DELETED_TCMD | 18 | | | | 6 | | | | 2 | | | | 176418 | | | |
| NXL1WQ | 112_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 113_SELECT_L7_MOVEMENT_CONTROL | 8 | | | | 11 | | | | 2 | | | | 0 | | | |
| NXL1WQ | 1131_FORMAT_L7_MOVEMENT | 19 | | | | 6 | | | | 3 | | | | 536256 | | | |
| NXL1WQ | 113_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 114_SELECT_L9_MOVEMENT_CONTROL | 6 | | | | 6 | | | | 2 | | | | 0 | | | |
| NXL1WQ | 1151_FORMAT_L9_MOVEMENT | 15 | | | | 5 | | | | 3 | | | | 77760 | | | |
| NXL1WQ | 114_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 115_SELECT_LB_MOVEMENT_CONTROL | 14 | | | | 13 | | | | 2 | | | | 0 | | | |
| NXL1WQ | 1151_FORMAT_LB_MOVEMENT | 15 | | | | 3 | | | | 2 | | | | 365040 | | | |
| NXL1WQ | 115_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 116_SELECT_LE_MOVEMENT_CONTROL | 8 | | | | 5 | | | | 2 | | | | 0 | | | |
| NXL1WQ | 1161_FORMAT_LE_PALLET_MOVEMENT | 13 | | | | 3 | | | | 2 | | | | 130000 | | | |
| NXL1WQ | 116_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 117_SELECT_LF_MOVEMENT_CONTROL | 12 | | | | 8 | | | | 2 | | | | 0 | | | |
| NXL1WQ | 1171_FORMAT_LF_MOVEMENT | 13 | | | | 3 | | | | 2 | | | | 63700 | | | |
| NXL1WQ | 117_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 118_SELECT_LH_MOVEMENT_CONTROL | 8 | | | | 6 | | | | 2 | | | | 288 | | | |
| NXL1WQ | 1181_FORMAT_LH_PALLET_MOVEMENT | 10 | | | | 3 | | | | 2 | | | | 24010 | | | |
| NXL1WQ | 118_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 119_SELECT_S3_MOVEMENT_CONTROL | 14 | | | | 16 | | | | 2 | | | | 0 | | | |
| NXL1WQ | 1191_FORMAT_S3_MOVEMENT | 12 | | | | 3 | | | | 2 | | | | 34992 | | | |
| NXL1WQ | 119_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 11A_SELECT_LP_MOVEMENT_CONTROL | 14 | | | | 12 | | | | 3 | | | | 0 | | | |
| NXL1WQ | 11A1_FORMAT_LP_MOVEMENT | 7 | | | | 2 | | | | 1 | | | | 9072 | | | |
| NXL1WQ | 11A_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 120_PROCESS_ON_HAND | 44 | | | | 16 | | | | 2 | | | | 10392624 | | | |
| NXL1WQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 121_READ_ON_HAND | 11 | | | | 4 | | | | 2 | | | | 9900 | | | |
| NXL1WQ | 121_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 122_FORMAT_L5_ON_HAND | 28 | | | | 19 | | | | 2 | | | | 1367548 | | | |
| NXL1WQ | 122_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 123_FORMAT_LD_ON_HAND | 20 | | | | 18 | | | | 2 | | | | 141120 | | | |
| NXL1WQ | 123_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 124_FORMAT_S1_ON_HAND | 22 | | | | 21 | | | | 2 | | | | 301158 | | | |
| NXL1WQ | 124_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 125_FORMAT_LF_ON_HAND | 21 | | | | 12 | | | | 2 | | | | 302400 | | | |
| NXL1WQ | 125_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 126_FORMAT_S3_ON_HAND | 20 | | | | 22 | | | | 2 | | | | 233280 | | | |
| NXL1WQ | 126_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 127_FORMAT_L4_ON_HAND | 32 | | | | 20 | | | | 2 | | | | 1952288 | | | |
| NXL1WQ | 127_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 128_FORMAT_LM_ON_HAND | 23 | | | | 34 | | | | 3 | | | | 190463 | | | |
| NXL1WQ | 128_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 129_FORMAT_LN_ON_HAND | 30 | | | | 50 | | | | 2 | | | | 546750 | | | |
| NXL1WQ | 129_OFFSHORE_CONTINUE | 27 | | | | 27 | | | | 3 | | | | 164268 | | | |
| NXL1WQ | 129_PROCESS_CONTINUE | 6 | | | | 5 | | | | 2 | | | | 864 | | | |
| NXL1WQ | 129_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 12A_FORMAT_LO_ON_HAND | 21 | | | | 17 | | | | 3 | | | | 127764 | | | |
| NXL1WQ | 12A_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 200_LOAD_FILES_DRIVER | 21 | | | | 11 | | | | 2 | | | | 134400 | | | |
| NXL1WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 210_ALLOCATE_AKF_FILE | 23 | | | | 12 | | | | 2 | | | | 476928 | | | |
| NXL1WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 220_AKF_FILE_BUILD | 8 | | | | 5 | | | | 2 | | | | 1152 | | | |
| NXL1WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | January | April | Baseline | October | January | April | Baseline | October | January | April | Baseline | October | January | April |
| NXL1WQ | 230_RELEASE_AKF_FILE | 11 | | | | 4 | | | | 2 | | | | 53900 | | | |
| NXL1WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 240_AKF_CLEANUP | 22 | | | | 11 | | | | 2 | | | | 360448 | | | |
| NXL1WQ | 240_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 300_REBUILD_PROCESS | 21 | | | | 6 | | | | 2 | | | | 18900 | | | |
| NXL1WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 310_FIND_RC | 5 | | | | 3 | | | | 2 | | | | 45 | | | |
| NXL1WQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 320_LOAD_TEMP_AKF | 8 | | | | 4 | | | | 2 | | | | 512 | | | |
| NXL1WQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 700_ADDFIL_ERROR_CHECK | 12 | | | | 4 | | | | 2 | | | | 43200 | | | |
| NXL1WQ | 700_ADDFIL_ERROR_CHECK_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 700_SEARCH_PGM_STA_TABLE | 6 | | | | 6 | | | | 2 | | | | 1176 | | | |
| NXL1WQ | 700_SEARCH_PGM_STA_TABLE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 700_SEARCH_CHNL_TABLE_DTL | 11 | | | | 7 | | | | 2 | | | | 14256 | | | |
| NXL1WQ | 700_SEARCH_CHNL_TABLE_DTL_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 700_SEARCH_CHNL_TABLE_PLT | 11 | | | | 7 | | | | 2 | | | | 14256 | | | |
| NXL1WQ | 700_SEARCH_CHNL_TABLE_PLT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 700_CHECK_NAF_INDICATORS | 18 | | | | 5 | | | | 2 | | | | 10368 | | | |
| NXL1WQ | 700_CHECK_NAF_INDICATORS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL1WQ | 700_FORMAT_NAF | 24 | | | | 25 | | | | 2 | | | | 13824 | | | |
| NXL1WQ | 700_FORMAT_NAF_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 000_L2_DRIVER | 18 | | | | 9 | | | | 2 | | | | 194688 | | | |
| NXL2WQ | 000_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 100_SLEW_TRANSLATOR_HSKP | 22 | | | | 3 | | | | 2 | | | | 285912 | | | |
| NXL2WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 200_SLEW_TRANSLATOR_DRIVER | 5 | | | | 2 | | | | 2 | | | | 80 | | | |
| NXL2WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 210_READ_L2_REPORT_FILE | 10 | | | | 4 | | | | 3 | | | | 1000 | | | |
| NXL2WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 220_TRANSLATE_AND_WRITE | 50 | | | | 13 | | | | 4 | | | | 520200 | | | |
| NXL2WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 221_SPACE_LINE | 5 | | | | 2 | | | | 1 | | | | 2205 | | | |
| NXL2WQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 222_REPORT_DATA | 7 | | | | 3 | | | | 2 | | | | 16128 | | | |
| NXL2WQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 300_SLEW_TRANSLATOR_WRAPUP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 400_RPT_TRANSMITTER_HSKP | 22 | | | | 2 | | | | 2 | | | | 400950 | | | |
| NXL2WQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 500_RPT_TRANSMITTER_DRIVER | 9 | | | | 8 | | | | 2 | | | | 8100 | | | |
| NXL2WQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 510_MINIS_SEND_DRIVER | 11 | | | | 8 | | | | 2 | | | | 6875 | | | |
| NXL2WQ | 510_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 511_FIRST_READ | 13 | | | | 3 | | | | 2 | | | | 15925 | | | |
| NXL2WQ | 511_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 512_LOAD_MINI_PRT_HDR | 13 | | | | 3 | | | | 2 | | | | 46800 | | | |
| NXL2WQ | 512_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 513_LOAD_REPORT_DATA | 19 | | | | 9 | | | | 2 | | | | 134064 | | | |
| NXL2WQ | 513_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 514_MESSAGE_CONTROL | 16 | | | | 12 | | | | 2 | | | | 9216 | | | |
| NXL2WQ | 514_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 515_FORMAT_CALL_RBTCHO | 22 | | | | 5 | | | | 2 | | | | 521752 | | | |
| NXL2WQ | 515_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 520_BD700_SEND_DRIVER | 3 | | | | 2 | | | | 1 | | | | 48 | | | |
| NXL2WQ | 520_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 521_FIRST_READ | 12 | | | | 2 | | | | 1 | | | | 14700 | | | |
| NXL2WQ | 521_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 522_READ_SEND_RECORD | 13 | | | | 3 | | | | 2 | | | | 20800 | | | |
| NXL2WQ | 522_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 600_RPT_TRANSMITTER_WRAPUP | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 600_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 700_ERROR_DISPLAY | 12 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL2WQ | 700_ERROR_DISPLAY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 000_L4_DRIVER | 13 | | | | 6 | | | | 2 | | | | 37908 | | | |
| NXL4WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXL4WQ | 100_HSKP | 25 | | | | 3 | | | | 1 | | | | 8410000 | | | |
| NXL4WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 110_LOAD_MAI_TABLE | 5 | | | | 2 | | | | 1 | | | | 405 | | | |
| NXL4WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 200_BUILD_REPORT | 10 | | | | 9 | | | | 2 | | | | 43560 | | | |
| NXL4WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 210_PROCESS_APOD | 29 | | | | 17 | | | | 2 | | | | 3025541 | | | |
| NXL4WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 211_SUPER_PRI_PROC | 24 | | | | 8 | | | | 2 | | | | 1944 | | | |
| NXL4WQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 212_PRI_1_2_3_PROC | 24 | | | | 8 | | | | 2 | | | | 1944 | | | |
| NXL4WQ | 212_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 213_TOTAL_PROC | 24 | | | | 8 | | | | 2 | | | | 1944 | | | |
| NXL4WQ | 213_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 214_PRI_4_PROC | 24 | | | | 8 | | | | 2 | | | | 1944 | | | |
| NXL4WQ | 214_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 215_COMPUTE_WRITE | 17 | | | | 7 | | | | 2 | | | | 27200 | | | |
| NXL4WQ | 215_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 220_PROCESS_SUB_AREA | 14 | | | | 6 | | | | 2 | | | | 140000 | | | |
| NXL4WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 221_UPDATE_SUB_AREA_TABLE | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL4WQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 2211_PRI_PROC | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL4WQ | 2211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 22111_AGE_PROC | 4 | | | | 1 | | | | 1 | | | | 3600 | | | |
| NXL4WQ | 22111_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 230_PROCESS_MAI_AREA | 19 | | | | 8 | | | | 3 | | | | 331056 | | | |
| NXL4WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 231_UPDATE_MAI_AREA_TABLE | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL4WQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 2311_PRI_PROC | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL4WQ | 2311_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 23111_AGE_PROC | 4 | | | | 1 | | | | 1 | | | | 3600 | | | |
| NXL4WQ | 23111_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 240_PROCESS_MFST_STATION | 73 | | | | 14 | | | | 2 | | | | 22892800 | | | |
| NXL4WQ | 240_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 241_HIGH_VOLUME | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL4WQ | 241_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 2411_PRI_PROC | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL4WQ | 2411_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 24111_AGE_PROC | 14 | | | | 2 | | | | 2 | | | | 17150 | | | |
| NXL4WQ | 24111_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 242_LOW_VOLUME | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL4WQ | 242_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 2421_PRI_PROC | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL4WQ | 2421_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 24211_AGE_PROC | 14 | | | | 2 | | | | 2 | | | | 17150 | | | |
| NXL4WQ | 24211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 243_VALIDATED_FREQ | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL4WQ | 243_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 2431_PRI_PROC | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL4WQ | 2431_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 24311_AGE_PROC | 14 | | | | 2 | | | | 2 | | | | 17150 | | | |
| NXL4WQ | 24311_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 244_OTHER | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL4WQ | 244_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 2441_PRI_PROC | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL4WQ | 2441_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 24411_AGE_PROC | 14 | | | | 2 | | | | 2 | | | | 17150 | | | |
| NXL4WQ | 24411_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 245_GRAND_TOTAL | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL4WQ | 245_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 2451_PRI_PROC | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL4WQ | 2451_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 24511_AGE_PROC | 21 | | | | 6 | | | | 2 | | | | 52500 | | | |
| NXL4WQ | 24511_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXL4WQ | 250_WRITE_GRAND_TOTALS | 4 | | | | 1 | | | | 1 | | | | 36 | | | |
| NXL4WQ | 250_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 251_WRITE_21ST_AF_TOTALS | 54 | | | | 11 | | | | 1 | | | | 857304 | | | |
| NXL4WQ | 251_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 252_WRITE_22ND_AF_TOTALS | 53 | | | | 11 | | | | 1 | | | | 750533 | | | |
| NXL4WQ | 252_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 253_WRITE_MAC_HQ_TOTALS | 51 | | | | 11 | | | | 1 | | | | 722211 | | | |
| NXL4WQ | 253_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 300_WRAPUP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 700_TRAILER | 6 | | | | 2 | | | | 1 | | | | 25350 | | | |
| NXL4WQ | 700_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 700_HEADERS_PART_I | 22 | | | | 12 | | | | 2 | | | | 1064800 | | | |
| NXL4WQ | 700_HEADERS_PART_I_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 700_HEADERS_PART_II | 9 | | | | 7 | | | | 1 | | | | 5184 | | | |
| NXL4WQ | 700_HEADERS_PART_II_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 700_WRITE_AF_TOTAL_HDR | 8 | | | | 7 | | | | 1 | | | | 2048 | | | |
| NXL4WQ | 700_WRITE_AF_TOTAL_HDR_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 700_WRITE_MAC_TOTAL_HDR | 8 | | | | 7 | | | | 1 | | | | 1568 | | | |
| NXL4WQ | 700_WRITE_MAC_TOTAL_HDR_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 700_ERROR_ROUTINE | 6 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 700_ERROR_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 700_COMPUTE | 3 | | | | 1 | | | | 1 | | | | 8112 | | | |
| NXL4WQ | 700_COMPUTE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 710_COMPUTE_TABLE | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL4WQ | 710_COMPUTE_TABLE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 711_PRI_PROC | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL4WQ | 711_PRI_PROC_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 7111_AGE_PROC | 4 | | | | 1 | | | | 1 | | | | 2304 | | | |
| NXL4WQ | 7111_TYPE_AGE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL4WQ | 720_FORMAT_WRITE | 77 | | | | 7 | | | | 1 | | | | 1509200 | | | |
| NXL4WQ | 720_FORMAT_WRITE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 000_L5_DRIVER | 13 | | | | 6 | | | | 2 | | | | 37908 | | | |
| NXL5WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 100_HSKP | 26 | | | | 3 | | | | 1 | | | | 9360000 | | | |
| NXL5WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 110_LOAD_MAI_TABLE | 6 | | | | 2 | | | | 1 | | | | 1350 | | | |
| NXL5WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 200_BUILD_REPORT | 10 | | | | 8 | | | | 2 | | | | 43560 | | | |
| NXL5WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 210_PROCESS_APOD | 38 | | | | 18 | | | | 3 | | | | 12563750 | | | |
| NXL5WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 211_SUPER_PRI_PROC | 24 | | | | 7 | | | | 2 | | | | 1944 | | | |
| NXL5WQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 212_PRI_1_2_3_PROC | 24 | | | | 7 | | | | 2 | | | | 1944 | | | |
| NXL5WQ | 212_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 213_TOTAL_PROC | 24 | | | | 7 | | | | 2 | | | | 1944 | | | |
| NXL5WQ | 213_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 214_PRI_4_PROC | 24 | | | | 7 | | | | 2 | | | | 1944 | | | |
| NXL5WQ | 214_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 215_COMPUTE_WRITE | 17 | | | | 7 | | | | 2 | | | | 27200 | | | |
| NXL5WQ | 215_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 220_PROCESS_SUB_AREA | 14 | | | | 6 | | | | 2 | | | | 140000 | | | |
| NXL5WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 221_UPDATE_SUB_AREA_TABLE | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL5WQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 2211_PRI_PROC | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL5WQ | 2211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 22111_AGE_PROC | 4 | | | | 1 | | | | 1 | | | | 3600 | | | |
| NXL5WQ | 22111_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 230_PROCESS_MAI_AREA | 18 | | | | 8 | | | | 2 | | | | 313632 | | | |
| NXL5WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 231_UPDATE_MAI_AREA_TABLE | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL5WQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 2311_PRI_PROC | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL5WQ | 2311_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXL5WQ | 23111_AGE_PROC | 4 | | | | 1 | | | | 1 | | | | 3600 | | | |
| NXL5WQ | 23111_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 240_PROCESS_MFST_STATION | 73 | | | | 14 | | | | 2 | | | | 22892800 | | | |
| NXL5WQ | 240_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 241_HIGH_VOLUME | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL5WQ | 241_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 2411_PRI_PROC | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL5WQ | 2411_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 24111_AGE_PROC | 14 | | | | 2 | | | | 2 | | | | 17150 | | | |
| NXL5WQ | 24111_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 242_LOW_VOLUME | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL5WQ | 242_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 2421_PRI_PROC | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL5WQ | 2421_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 24211_AGE_PROC | 14 | | | | 2 | | | | 2 | | | | 17150 | | | |
| NXL5WQ | 24211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 243_VALIDATED_FREQ | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL5WQ | 243_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 2431_PRI_PROC | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL5WQ | 2431_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 24311_AGE_PROC | 14 | | | | 2 | | | | 2 | | | | 17150 | | | |
| NXL5WQ | 24311_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 244_OTHER | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL5WQ | 244_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 2441_PRI_PROC | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL5WQ | 2441_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 24411_AGE_PROC | 14 | | | | 2 | | | | 2 | | | | 17150 | | | |
| NXL5WQ | 24411_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 245_GRAND_TOTAL | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL5WQ | 245_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 2451_PRI_PROC | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL5WQ | 2451_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 24511_AGE_PROC | 21 | | | | 6 | | | | 2 | | | | 52500 | | | |
| NXL5WQ | 24511_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 250_WRITE_GRAND_TOTALS | 4 | | | | 1 | | | | 1 | | | | 36 | | | |
| NXL5WQ | 250_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 251_WRITE_21ST_AF_TOTALS | 54 | | | | 11 | | | | 1 | | | | 857304 | | | |
| NXL5WQ | 251_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 252_WRITE_22ND_AF_TOTALS | 53 | | | | 11 | | | | 1 | | | | 750533 | | | |
| NXL5WQ | 252_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 253_WRITE_MAC_HQ_TOTALS | 51 | | | | 11 | | | | 1 | | | | 722211 | | | |
| NXL5WQ | 253_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 300_WRAPUP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 700_TRAILER | 6 | | | | 2 | | | | 1 | | | | 25350 | | | |
| NXL5WQ | 700_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 700_HEADERS_PART_I | 22 | | | | 12 | | | | 2 | | | | 1163800 | | | |
| NXL5WQ | 700_HEADERS_PART_I_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 700_HEADERS_PART_II | 9 | | | | 7 | | | | 1 | | | | 5184 | | | |
| NXL5WQ | 700_HEADERS_PART_II_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 700_WRITE_AF_TOTAL_HDR | 8 | | | | 7 | | | | 1 | | | | 2048 | | | |
| NXL5WQ | 700_WRITE_AF_TOTAL_HDR_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 700_WRITE_MAC_TOTAL_HDR | 8 | | | | 7 | | | | 1 | | | | 1568 | | | |
| NXL5WQ | 700_WRITE_MAC_TOTAL_HDR_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 700_ERROR_ROUTINE | 6 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 700_ERROR_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 700_COMPUTE | 3 | | | | 1 | | | | 1 | | | | 8112 | | | |
| NXL5WQ | 700_COMPUTE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 710_COMPUTE_TABLE | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL5WQ | 710_COMPUTE_TABLE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 711_PRI_PROC | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXL5WQ | 711_PRI_PROC_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 7111_AGE_PROC | 4 | | | | 1 | | | | 1 | | | | 2304 | | | |
| NXL5WQ | 7111_TYPE_AGE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL5WQ | 720_FORMAT_WRITE | 82 | | | | 7 | | | | 1 | | | | 1944712 | | | |
| NXL5WQ | 720_FORMAT_WRITE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXL7WQ | 000_DRIVER | 12 | | | | 4 | | | | 2 | | | | 10800 | | | |
| NXL7WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL7WQ | 100_HOUSEKEEPING | 26 | | | | 4 | | | | 1 | | | | 5890976 | | | |
| NXL7WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL7WQ | 110_FORMAT_TOP_HDR | 12 | | | | 3 | | | | 1 | | | | 1728 | | | |
| NXL7WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL7WQ | 200_BUILD_REPORT | 12 | | | | 6 | | | | 2 | | | | 2700 | | | |
| NXL7WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL7WQ | 210_READ_AKF | 5 | | | | 2 | | | | 1 | | | | 45 | | | |
| NXL7WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL7WQ | 220_PROCESS_DATA | 20 | | | | 6 | | | | 2 | | | | 87120 | | | |
| NXL7WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL7WQ | 221_READ_MOVEMENT | 8 | | | | 2 | | | | 1 | | | | 1152 | | | |
| NXL7WQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL7WQ | 222_FORMAT_DETAIL | 49 | | | | 16 | | | | 2 | | | | 46294416 | | | |
| NXL7WQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL7WQ | 222_A_CHK_ZERO_HR | 3 | | | | 3 | | | | 2 | | | | 27 | | | |
| NXL7WQ | 222_A_CHK_ZERO_HR_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL7WQ | 222_T_CHK_ZERO_HR | 3 | | | | 3 | | | | 2 | | | | 27 | | | |
| NXL7WQ | 222_T_CHK_ZERO_HR_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL7WQ | 223_WRITE_PAGE_HEADER | 11 | | | | 5 | | | | 2 | | | | 89100 | | | |
| NXL7WQ | 223_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL7WQ | 224_WRITE_SUB_HEADER | 22 | | | | 7 | | | | 2 | | | | 68992 | | | |
| NXL7WQ | 224_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL7WQ | 225_WRITE_DETAIL | 10 | | | | 4 | | | | 2 | | | | 6250 | | | |
| NXL7WQ | 225_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL7WQ | 226_WRITE_PAGE_TRAILER | 5 | | | | 2 | | | | 1 | | | | 3920 | | | |
| NXL7WQ | 226_WRITE_PAGE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL7WQ | 300_WRAP_UP | 4 | | | | 1 | | | | 1 | | | | 256 | | | |
| NXL7WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 000_L9_CONTROL_DRIVER | 13 | | | | 6 | | | | 2 | | | | 11700 | | | |
| NXL9WQ | 000_DRIVER_COMPLETE | 4 | | | | 2 | | | | 2 | | | | 196 | | | |
| NXL9WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 100_HSKP_DRIVER | 28 | | | | 6 | | | | 2 | | | | 5347132 | | | |
| NXL9WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 200_UPDATE_DRIVER | 40 | | | | 29 | | | | 4 | | | | 585640 | | | |
| NXL9WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 210_CREATE | 7 | | | | 5 | | | | 2 | | | | 5488 | | | |
| NXL9WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 220_COPY | 5 | | | | 4 | | | | 2 | | | | 1620 | | | |
| NXL9WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 230_UPDATE | 38 | | | | 21 | | | | 3 | | | | 4290048 | | | |
| NXL9WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 231_SUPER_PRIORITY_ADD | 6 | | | | 2 | | | | 2 | | | | 2400 | | | |
| NXL9WQ | 231_ADD_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 231_SUPER_PRIORITY_SUBTRACT | 10 | | | | 2 | | | | 2 | | | | 16000 | | | |
| NXL9WQ | 231_SUBT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 232_PRIORITY_1_2_3_ADD | 6 | | | | 2 | | | | 2 | | | | 2400 | | | |
| NXL9WQ | 232_ADD_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 232_PRIORITY_1_2_3_SUBTRACT | 10 | | | | 2 | | | | 2 | | | | 16000 | | | |
| NXL9WQ | 232_SUBT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 233_SUPER_THRU_3_ADD | 7 | | | | 2 | | | | 2 | | | | 5488 | | | |
| NXL9WQ | 233_ADD_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 233_SUPER_THRU_3_SUBTRACT | 11 | | | | 2 | | | | 2 | | | | 34496 | | | |
| NXL9WQ | 233_SUBT_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| NXL9WQ | 234_PRIORITY_4_ADD | 6 | | | | 2 | | | | 2 | | | | 2400 | | | |
| NXL9WQ | 234_ADD_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 234_PRIORITY_4_SUBTRACT | 10 | | | | 2 | | | | 2 | | | | 16000 | | | |
| NXL9WQ | 234_SUBT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 300_BUILD_REPORT_DRIVER | 35 | | | | 8 | | | | 2 | | | | 2551500 | | | |
| NXL9WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 310_TYPE_PROCESS | 84 | | | | 7 | | | | 3 | | | | 261382464 | | | |
| NXL9WQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 311_READ_NEW_SUBT | 6 | | | | 3 | | | | 2 | | | | 864 | | | |
| NXL9WQ | 311_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 320_SUB_TOTAL_OUTPUT | 58 | | | | 8 | | | | 2 | | | | 53452800 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXL9WQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 330_STN_TOTAL_OUTPUT | 39 | | | | 5 | | | | 2 | | | | 11883456 | | | |
| NXL9WQ | 330_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 400_WRAP_UP | 5 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXL9WQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 700_WRITE_TRAILER | 7 | | | | 3 | | | | 2 | | | | 17500 | | | |
| NXL9WQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 700_WRITE_HEADA | 4 | | | | 3 | | | | 2 | | | | 1024 | | | |
| NXL9WQ | 700_WRITE_HEADA_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXL9WQ | 700_WRITE_SUB_HEADINGS | 18 | | | | 11 | | | | 2 | | | | 101250 | | | |
| NXL9WQ | 700_WRITE_SUB_HEADINGS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 000_DRIVER | 18 | | | | 9 | | | | 2 | | | | 259200 | | | |
| NXLBWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 100_HOUSEKEEPING | 48 | | | | 10 | | | | 3 | | | | 67972800 | | | |
| NXLBWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 110_LOAD_MAI_TABLE | 6 | | | | 2 | | | | 1 | | | | 1350 | | | |
| NXLBWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 200_UPDATE_DRIVER | 6 | | | | 4 | | | | 2 | | | | 384 | | | |
| NXLBWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 210_READ_CONTROL | 8 | | | | 5 | | | | 2 | | | | 1800 | | | |
| NXLBWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 211_READ_AKF | 7 | | | | 3 | | | | 2 | | | | 1575 | | | |
| NXLBWQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 212_READ_DAILY_MOVEMENT | 12 | | | | 4 | | | | 2 | | | | 9408 | | | |
| NXLBWQ | 212_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 213_READ_OLD_CUM_MVMT | 10 | | | | 3 | | | | 2 | | | | 12960 | | | |
| NXLBWQ | 213_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 220_UPDATE_AND_WRITE_CONTROL | 14 | | | | 6 | | | | 2 | | | | 12600 | | | |
| NXLBWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 221_WRITE_CUM_AND_DAILY_REC | 12 | | | | 4 | | | | 2 | | | | 37632 | | | |
| NXLBWQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 222_CREATE_NEW_CUM_REC | 87 | | | | 47 | | | | 3 | | | | 74278512 | | | |
| NXLBWQ | 222_CONTINUE | 16 | | | | 6 | | | | 2 | | | | 46656 | | | |
| NXLBWQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 223_UPDATE_CUM_REC | 156 | | | | 54 | | | | 4 | | | | 215744256 | | | |
| NXLBWQ | 223_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 300_REPORT_DRIVER | 6 | | | | 3 | | | | 2 | | | | 486 | | | |
| NXLBWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 310_READ_DAILY_AND_CUM | 11 | | | | 4 | | | | 2 | | | | 26411 | | | |
| NXLBWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 320_FORMAT_AND_WRITE_CONTROL | 12 | | | | 9 | | | | 2 | | | | 24300 | | | |
| NXLBWQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 321_SUB_AREA_PROCESS | 14 | | | | 4 | | | | 2 | | | | 68600 | | | |
| NXLBWQ | 321_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 322_MAI_PROCESS | 16 | | | | 5 | | | | 2 | | | | 123904 | | | |
| NXLBWQ | 322_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 323_STATION_PROCESS | 96 | | | | 11 | | | | 2 | | | | 3538944 | | | |
| NXLBWQ | 323_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 324_APOD_PROCESS | 53 | | | | 15 | | | | 4 | | | | 2588573 | | | |
| NXLBWQ | 324_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 330_WRITE_GRAND_TOTALS | 5 | | | | 2 | | | | 2 | | | | 180 | | | |
| NXLBWQ | 330_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 331_WRITE_21ST_AF_TOTALS | 44 | | | | 7 | | | | 2 | | | | 228096 | | | |
| NXLBWQ | 331_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 332_WRITE_22ND_AF_TOTALS | 44 | | | | 7 | | | | 2 | | | | 228096 | | | |
| NXLBWQ | 332_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 333_WRITE_AMC_HQ_TOTALS | 57 | | | | 7 | | | | 2 | | | | 295488 | | | |
| NXLBWQ | 333_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 400_WRAPUP | 3 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXLBWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 700_COMPUTE_PROCESS | 197 | | | | 15 | | | | 2 | | | | 3591881300 | | | |
| NXLBWQ | 700_COMPUTE_PROCESS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 700_WRITE_HEADER_PART_1 | 18 | | | | 10 | | | | 3 | | | | 130050 | | | |
| NXLBWQ | 700_WRITE_HEADER_PART1_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 700_WRITE_HEADER_PART_2 | 9 | | | | 8 | | | | 2 | | | | 2916 | | | |
| NXLBWQ | 700_WRITE_HEADER_PART2_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXLBWQ | 700_WRITE_AF_TOTAL_HDR | 9 | | | | 8 | | | | 2 | | | | 2916 | | | |
| NXLBWQ | 700_WRITE_AF_TOTAL_HDR_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 700_WRITE_AMC_TOTAL_HDR | 9 | | | | 8 | | | | 2 | | | | 2304 | | | |
| NXLBWQ | 700_WRITE_AMC_TOTAL_HDR_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 700_WRITE_TRAILER | 14 | | | | 5 | | | | 2 | | | | 229376 | | | |
| NXLBWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLBWQ | 700_SEARCH_MAI_CODE | 6 | | | | 4 | | | | 2 | | | | 1176 | | | |
| NXLBWQ | 700_SEARCH_MAI_CODE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLDWQ | 000_LD_DRIVER | 15 | | | | 7 | | | | 2 | | | | 73500 | | | |
| NXLDWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLDWQ | 100_HSKP | 18 | | | | 3 | | | | 2 | | | | 1312200 | | | |
| NXLDWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLDWQ | 200_BUILD_REPORT | 18 | | | | 5 | | | | 3 | | | | 472392 | | | |
| NXLDWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLDWQ | 210_READ_LD_AKF | 4 | | | | 3 | | | | 2 | | | | 36 | | | |
| NXLDWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLDWQ | 220_PROCESS_REPORT_DATA | 15 | | | | 5 | | | | 2 | | | | 54000 | | | |
| NXLDWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLDWQ | 221_WRITE_CHANNEL_LINE | 30 | | | | 6 | | | | 2 | | | | 14532480 | | | |
| NXLDWQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLDWQ | 222_WRITE_GRAND_TOTALS | 12 | | | | 5 | | | | 2 | | | | 62208 | | | |
| NXLDWQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLDWQ | 300_WRAPUP | 3 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXLDWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLDWQ | 700_READ_AND_ADD | 13 | | | | 3 | | | | 2 | | | | 37908 | | | |
| NXLDWQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLDWQ | 700_WRITE_TRAILER | 7 | | | | 3 | | | | 2 | | | | 11200 | | | |
| NXLDWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLDWQ | 700_WRITE_HEADERS | 11 | | | | 7 | | | | 2 | | | | 5324 | | | |
| NXLDWQ | 700_WRITE_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLDWQ | 700_ERROR_ROUTINE | 7 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXLDWQ | 700_ERROR_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 000_LE_DRIVER | 13 | | | | 6 | | | | 2 | | | | 37908 | | | |
| NXLEWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 100_HOUSEKEEPING | 22 | | | | 2 | | | | 1 | | | | 2979328 | | | |
| NXLEWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 200_BUILD_REPORT | 7 | | | | 1 | | | | 1 | | | | 252 | | | |
| NXLEWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 210_READ_DETAIL_AND_NEW_AKF | 28 | | | | 5 | | | | 3 | | | | 3568572 | | | |
| NXLEWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 220_PROCESS_PRIORITY_DATA | 21 | | | | 25 | | | | 3 | | | | 108864 | | | |
| NXLEWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 230_PROCESS_DETAIL | 21 | | | | 19 | | | | 3 | | | | 571725 | | | |
| NXLEWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 231_WRITE_DETAIL | 36 | | | | 7 | | | | 2 | | | | 33246756 | | | |
| NXLEWQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 240_PROCESS_MANIFEST_DEST | 25 | | | | 17 | | | | 3 | | | | 1440000 | | | |
| NXLEWQ | 240_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 241_WRITE_MANIFEST_DEST | 20 | | | | 4 | | | | 1 | | | | 1479680 | | | |
| NXLEWQ | 241_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 250_PROCESS_MISSION | 27 | | | | 16 | | | | 3 | | | | 1529388 | | | |
| NXLEWQ | 250_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 251_WRITE_PRI1_3_TOTAL | 14 | | | | 2 | | | | 1 | | | | 129024 | | | |
| NXLEWQ | 251_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 252_WRITE_TP4_TOTAL | 19 | | | | 3 | | | | 2 | | | | 393984 | | | |
| NXLEWQ | 252_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 253_WRITE_MISSION | 23 | | | | 5 | | | | 2 | | | | 1907712 | | | |
| NXLEWQ | 253_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 260_PROCESS_STATION | 40 | | | | 17 | | | | 3 | | | | 14113440 | | | |
| NXLEWQ | 260_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 261_WRITE_STATION | 18 | | | | 3 | | | | 1 | | | | 911250 | | | |
| NXLEWQ | 261_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 300_WRAPUP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 700_WRITE_TRAILER | 6 | | | | 2 | | | | 1 | | | | 15000 | | | |
| NXLEWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXLEWQ | 700_WRITE_HEADERS | 9 | | | | 7 | | | | 1 | | | | 8100 | | | |
| NXLEWQ | 700_WRITE_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 700_ERROR_ROUTINE | 6 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLEWQ | 700_ERROR_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 000_LF_DRIVER | 12 | | | | 7 | | | | 2 | | | | 47628 | | | |
| NXLFWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 100_HSKP | 62 | | | | 14 | | | | 3 | | | | 52934112 | | | |
| NXLFWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 110_LOAD_STATION_TABLE | 28 | | | | 11 | | | | 3 | | | | 100800 | | | |
| NXLFWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 120_RESEQUENCE_SP_A_FILE | 6 | | | | 3 | | | | 2 | | | | 600 | | | |
| NXLFWQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 121_PROCESS_SPACE_A_FILE | 9 | | | | 5 | | | | 2 | | | | 1296 | | | |
| NXLFWQ | 121_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 200_GEN_FILE_UPDATE | 49 | | | | 16 | | | | 4 | | | | 2480625 | | | |
| NXLFWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 210_UPDATE_WITH_TEMP_GEN_REC | 40 | | | | 3 | | | | 2 | | | | 7464960 | | | |
| NXLFWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 220_UPDATE_WITH_TC_REC | 14 | | | | 8 | | | | 3 | | | | 22400 | | | |
| NXLFWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 221_PROCESS_MVMT_LIFT_24 | 49 | | | | 20 | | | | 8 | | | | 1254400 | | | |
| NXLFWQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 222_PROCESS_MVMT_LIFT | 35 | | | | 20 | | | | 8 | | | | 283500 | | | |
| NXLFWQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 223_PROCESS_MVMT_SUBT | 35 | | | | 20 | | | | 8 | | | | 283500 | | | |
| NXLFWQ | 223_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 224_PROCESS_ONHAND | 35 | | | | 20 | | | | 8 | | | | 283500 | | | |
| NXLFWQ | 224_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 300_BUILD_REPORT | 8 | | | | 6 | | | | 2 | | | | 4608 | | | |
| NXLFWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 310_HSKP | 13 | | | | 3 | | | | 2 | | | | 63700 | | | |
| NXLFWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 320_BUILD_PORTS_REPORT | 14 | | | | 7 | | | | 2 | | | | 204974 | | | |
| NXLFWQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 321_PROCESS_GEN_REC | 7 | | | | 3 | | | | 2 | | | | 0 | | | |
| NXLFWQ | 321_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 322_PROCESS_SP_ASSIGN_REC | 7 | | | | 4 | | | | 3 | | | | 700 | | | |
| NXLFWQ | 322_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 3221_LOAD_SP_ASSIGN_REC | 51 | | | | 3 | | | | 2 | | | | 249900 | | | |
| NXLFWQ | 3221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 330_WRITE_HQS_TOTALS | 14 | | | | 2 | | | | 2 | | | | 6174 | | | |
| NXLFWQ | 330_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 340_DISPLAY_COMPUTES | 26 | | | | 2 | | | | 2 | | | | 1124864 | | | |
| NXLFWQ | 340_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 400_WRAPUP | 9 | | | | 3 | | | | 2 | | | | 9801 | | | |
| NXLFWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 700_ERROR_ROUTINE | 4 | | | | 2 | | | | 2 | | | | 256 | | | |
| NXLFWQ | 700_ERROR_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 700_SEARCH_STATION_TABLE | 8 | | | | 7 | | | | 2 | | | | 512 | | | |
| NXLFWQ | 700_SEARCH_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 700_READ_LF_AKF | 10 | | | | 5 | | | | 2 | | | | 1440 | | | |
| NXLFWQ | 700_READ_AKF_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 700_READ_TEMP_GEN_REC | 13 | | | | 5 | | | | 2 | | | | 5733 | | | |
| NXLFWQ | 700_READ_TEMP_GEN_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 700_READ_TC_OH_REC | 11 | | | | 4 | | | | 2 | | | | 3564 | | | |
| NXLFWQ | 700_READ_TC_OH_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 700_READ_TC_MV_REC | 12 | | | | 4 | | | | 2 | | | | 6912 | | | |
| NXLFWQ | 700_READ_TC_MV_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 700_READ_GEN_REC | 8 | | | | 4 | | | | 2 | | | | 1152 | | | |
| NXLFWQ | 700_READ_GEN_REC_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 700_READ_SPACE_A_FILE | 7 | | | | 4 | | | | 2 | | | | 448 | | | |
| NXLFWQ | 700_READ_SPACE_A_FILE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 700_WRITE_OUTPUT | 47 | | | | 18 | | | | 2 | | | | 5306112 | | | |
| NXLFWQ | 700_WRITE_OUTPUT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLFWQ | 700_CONVERT_DATA | 46 | | | | 3 | | | | 2 | | | | 111658744 | | | |
| NXLFWQ | 700_CONVERT_DATA_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXLFWQ | 700_LOAD_AS_OF_DATE | 15 | | | | 6 | | | | 2 | | | | 181500 | | | |
| NXLFWQ | 700_LOAD_AS_OF_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLGWQ | 000_DRIVER | 10 | | | | 5 | | | | 2 | | | | 17640 | | | |
| NXLGWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLGWQ | 100_HOUSEKEEPING | 27 | | | | 5 | | | | 3 | | | | 6750000 | | | |
| NXLGWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLGWQ | 200_PROCESS_RECORDS | 15 | | | | 7 | | | | 2 | | | | 73500 | | | |
| NXLGWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLGWQ | 210_PROCESS_NEW_MFST_STATION | 13 | | | | 4 | | | | 2 | | | | 46800 | | | |
| NXLGWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLGWQ | 220_WRITE_PART_TWO_HEADING | 13 | | | | 3 | | | | 2 | | | | 16848 | | | |
| NXLGWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLGWQ | 230_PRODUCE_DETAIL_LINE | 10 | | | | 4 | | | | 2 | | | | 7840 | | | |
| NXLGWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLGWQ | 231_FORMAT_DETAIL_LINE | 41 | | | | 3 | | | | 2 | | | | 40184100 | | | |
| NXLGWQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLGWQ | 300_WRAPUP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLGWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLGWQ | 700_READ_AKF_AND_PZ1 | 11 | | | | 3 | | | | 1 | | | | 6336 | | | |
| NXLGWQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLGWQ | 700_WRITE_TRAILER | 9 | | | | 3 | | | | 2 | | | | 20736 | | | |
| NXLGWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLGWQ | 700_WRITE_HEADINGS | 12 | | | | 8 | | | | 2 | | | | 6912 | | | |
| NXLGWQ | 700_WRITE_HEADINGS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLGWQ | 700_ERROR_ROUTINE | 6 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLGWQ | 700_ERROR_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLHWQ | 000_LH_DRIVER | 15 | | | | 7 | | | | 2 | | | | 73500 | | | |
| NXLHWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLHWQ | 100_HSKP | 34 | | | | 6 | | | | 3 | | | | 8636544 | | | |
| NXLHWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLHWQ | 200_PROCESS | 14 | | | | 3 | | | | 2 | | | | 42350 | | | |
| NXLHWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLHWQ | 210_PROCESS_DETAILS | 21 | | | | 7 | | | | 2 | | | | 162624 | | | |
| NXLHWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLHWQ | 211_FORMAT_REPORT | 36 | | | | 2 | | | | 2 | | | | 50979600 | | | |
| NXLHWQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLHWQ | 212_UPDATE_COUNTERS | 135 | | | | 38 | | | | 2 | | | | 34292160 | | | |
| NXLHWQ | 212_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLHWQ | 220_PROCESS_TOTALS | 4 | | | | 2 | | | | 2 | | | | 64 | | | |
| NXLHWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLHWQ | 221_FORMAT_TOTALS | 65 | | | | 2 | | | | 2 | | | | 1089986625 | | | |
| NXLHWQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLHWQ | 222_WRITE_TOTALS | 32 | | | | 31 | | | | 2 | | | | 115200 | | | |
| NXLHWQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLHWQ | 300_WRAPUP | 3 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXLHWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLHWQ | 700_WRITE_HEADER_PART_1 | 10 | | | | 8 | | | | 2 | | | | 25000 | | | |
| NXLHWQ | 700_WRITE_HEADER_PART1_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLHWQ | 700_WRITE_HEADER_PART_2 | 7 | | | | 6 | | | | 2 | | | | 1008 | | | |
| NXLHWQ | 700_WRITE_HEADER_PART2_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLHWQ | 700_WRITE_TRAILER | 7 | | | | 3 | | | | 2 | | | | 8575 | | | |
| NXLHWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLHWQ | 700_ERROR_ROUTINE | 5 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXLHWQ | 700_ERROR_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLMWQ | 000_LM_DRIVER | 15 | | | | 7 | | | | 2 | | | | 73500 | | | |
| NXLMWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLMWQ | 100_HSKP | 30 | | | | 4 | | | | 2 | | | | 20966880 | | | |
| NXLMWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLMWQ | 200_BUILD_REPORT | 4 | | | | 2 | | | | 2 | | | | 64 | | | |
| NXLMWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLMWQ | 210_READ_LM_AKF | 5 | | | | 3 | | | | 2 | | | | 180 | | | |
| NXLMWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLMWQ | 220_PROCESS_REPORT_DATA | 27 | | | | 10 | | | | 3 | | | | 789507 | | | |
| NXLMWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLMWQ | 221_WRITE_PRIORITY_LINE | 41 | | | | 13 | | | | 2 | | | | 14760000 | | | |
| NXLMWQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXLMWQ | 222_WRITE_TYPE_TOTALS | 29 | | | | 6 | | | | 2 | | | | 6189876 | | | |
| NXLMWQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLMWQ | 223_WRITE_STATION_TOTALS | 26 | | | | 5 | | | | 2 | | | | 5033600 | | | |
| NXLMWQ | 223_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLMWQ | 300_WRAPUP | 3 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXLMWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLMWQ | 700_READ_AND_ADD | 30 | | | | 13 | | | | 2 | | | | 702270 | | | |
| NXLMWQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLMWQ | 700_WRITE_TRAILER | 7 | | | | 3 | | | | 2 | | | | 11200 | | | |
| NXLMWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLMWQ | 700_WRITE_HEADERS | 8 | | | | 6 | | | | 2 | | | | 2592 | | | |
| NXLMWQ | 700_WRITE_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLMWQ | 700_ERROR_ROUTINE | 7 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXLMWQ | 700_ERROR_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLNWQ | 000_LN_DRIVER | 13 | | | | 6 | | | | 2 | | | | 37908 | | | |
| NXLNWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLNWQ | 100_HSKP | 30 | | | | 3 | | | | 2 | | | | 17146080 | | | |
| NXLNWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLNWQ | 200_BUILD_REPORT | 3 | | | | 1 | | | | 1 | | | | 12 | | | |
| NXLNWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLNWQ | 210_READ_LN_AKF | 4 | | | | 2 | | | | 1 | | | | 64 | | | |
| NXLNWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLNWQ | 220_PROCESS_REPORT_DATA | 24 | | | | 8 | | | | 2 | | | | 497664 | | | |
| NXLNWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLNWQ | 221_WRITE_CHANNEL_LINE | 33 | | | | 5 | | | | 2 | | | | 28174608 | | | |
| NXLNWQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLNWQ | 222_WRITE_TOTALS | 23 | | | | 10 | | | | 2 | | | | 150903 | | | |
| NXLNWQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLNWQ | 223_WRITE_GRAND_TOTALS | 12 | | | | 4 | | | | 2 | | | | 47628 | | | |
| NXLNWQ | 223_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLNWQ | 300_WRAPUP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLNWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLNWQ | 700_READ_AND_ADD | 15 | | | | 2 | | | | 1 | | | | 77760 | | | |
| NXLNWQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLNWQ | 700_WRITE_TRAILER | 6 | | | | 2 | | | | 1 | | | | 7350 | | | |
| NXLNWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLNWQ | 700_WRITE_HEADERS | 10 | | | | 6 | | | | 2 | | | | 7290 | | | |
| NXLNWQ | 700_WRITE_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLNWQ | 700_WRITE_DETAIL_HEADERS | 22 | | | | 11 | | | | 2 | | | | 12672 | | | |
| NXLNWQ | 700_WRITE_DETAIL_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLNWQ | 700_ERROR_ROUTINE | 6 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLNWQ | 700_ERROR_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLOWQ | 000_LO_DRIVER | 13 | | | | 6 | | | | 2 | | | | 37908 | | | |
| NXLOWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLOWQ | 100_HSKP | 18 | | | | 2 | | | | 1 | | | | 1462050 | | | |
| NXLOWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLOWQ | 200_BUILD_REPORT | 18 | | | | 4 | | | | 3 | | | | 373248 | | | |
| NXLOWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLOWQ | 210_READ_LO_AKF | 4 | | | | 2 | | | | 1 | | | | 64 | | | |
| NXLOWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLOWQ | 220_PROCESS_REPORT_DATA | 17 | | | | 6 | | | | 3 | | | | 88128 | | | |
| NXLOWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLOWQ | 221_WRITE_APOD_LINE | 19 | | | | 3 | | | | 2 | | | | 536256 | | | |
| NXLOWQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLOWQ | 222_WRITE_STATION_TOTALS | 19 | | | | 4 | | | | 2 | | | | 462384 | | | |
| NXLOWQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLOWQ | 230_WRITE_GRAND_TOTALS | 12 | | | | 2 | | | | 1 | | | | 78732 | | | |
| NXLOWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLOWQ | 300_WRAPUP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLOWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLOWQ | 700_READ_AND_ADD | 18 | | | | 3 | | | | 2 | | | | 139392 | | | |
| NXLOWQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLOWQ | 700_WRITE_TRAILER | 6 | | | | 2 | | | | 1 | | | | 7350 | | | |
| NXLOWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLOWQ | 700_WRITE_HEADERS | 8 | | | | 6 | | | | 1 | | | | 2592 | | | |
| NXLOWQ | 700_WRITE_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXLOWQ | 700_ERROR_ROUTINE | 6 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLOWQ | 700_ERROR_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLPWQ | 000_LP_DRIVER | 13 | | | | 6 | | | | 2 | | | | 37908 | | | |
| NXLPWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLPWQ | 100_HSKP | 18 | | | | 2 | | | | 1 | | | | 1462050 | | | |
| NXLPWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLPWQ | 200_BUILD_REPORT | 18 | | | | 4 | | | | 3 | | | | 373248 | | | |
| NXLPWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLPWQ | 210_READ_LP_AKF | 4 | | | | 2 | | | | 1 | | | | 64 | | | |
| NXLPWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLPWQ | 220_PROCESS_REPORT_DATA | 17 | | | | 6 | | | | 3 | | | | 88128 | | | |
| NXLPWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLPWQ | 221_WRITE_APOD_LINE | 14 | | | | 3 | | | | 2 | | | | 83006 | | | |
| NXLPWQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLPWQ | 222_WRITE_STATION_TOTALS | 15 | | | | 4 | | | | 2 | | | | 88935 | | | |
| NXLPWQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLPWQ | 230_WRITE_GRAND_TOTALS | 8 | | | | 2 | | | | 1 | | | | 6272 | | | |
| NXLPWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLPWQ | 300_WRAPUP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLPWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLPWQ | 700_READ_AND_ADD | 11 | | | | 2 | | | | 1 | | | | 13475 | | | |
| NXLPWQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLPWQ | 700_WRITE_TRAILER | 6 | | | | 2 | | | | 1 | | | | 7350 | | | |
| NXLPWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLPWQ | 700_WRITE_HEADERS | 8 | | | | 6 | | | | 1 | | | | 2592 | | | |
| NXLPWQ | 700_WRITE_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLPWQ | 700_ERROR_ROUTINE | 6 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLPWQ | 700_ERROR_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 000_MAIN_DRIVER | 13 | | | | 12 | | | | 2 | | | | 40768 | | | |
| NXLZWQ | 000_CONTINUE | 7 | | | | 3 | | | | 2 | | | | 1008 | | | |
| NXLZWQ | 000_STOP | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 100_HSKP | 61 | | | | 14 | | | | 3 | | | | 24985600 | | | |
| NXLZWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 110_LOAD_VALID_PORT_TABLE | 10 | | | | 4 | | | | 3 | | | | 2250 | | | |
| NXLZWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 200_L1_PROCESS | 6 | | | | 5 | | | | 2 | | | | 3456 | | | |
| NXLZWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 210_SEARCH_TABLE | 5 | | | | 4 | | | | 2 | | | | 180 | | | |
| NXLZWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 220_BUILD_MSG | 26 | | | | 10 | | | | 4 | | | | 842400 | | | |
| NXLZWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 300_Z4_PROCESS | 25 | | | | 15 | | | | 4 | | | | 302500 | | | |
| NXLZWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 310_CHECK_HISTORY | 18 | | | | 23 | | | | 4 | | | | 165888 | | | |
| NXLZWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 320_SEARCH_TABLE | 5 | | | | 4 | | | | 2 | | | | 180 | | | |
| NXLZWQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 330_BUILD_ADAM_III_MSG | 18 | | | | 10 | | | | 3 | | | | 56448 | | | |
| NXLZWQ | 330_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 331_BUILD_ADAM3_410_MSG_TABLE | 24 | | | | 5 | | | | 2 | | | | 921984 | | | |
| NXLZWQ | 331_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 3311_BUILD_ADAM3_410_DELE_MSG | 14 | | | | 3 | | | | 2 | | | | 72576 | | | |
| NXLZWQ | 3311_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 340_BUILD_ASIF_MSG | 17 | | | | 10 | | | | 3 | | | | 166617 | | | |
| NXLZWQ | 340_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 341_BUILD_ASIF_410_MSG_TABLE | 15 | | | | 4 | | | | 2 | | | | 77760 | | | |
| NXLZWQ | 341_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 400_L7_PROCESS | 24 | | | | 18 | | | | 3 | | | | 290400 | | | |
| NXLZWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 410_SEARCH_TABLE | 6 | | | | 4 | | | | 2 | | | | 216 | | | |
| NXLZWQ | 410_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 420_BUILD_ADAM_III_MSG | 18 | | | | 9 | | | | 3 | | | | 56448 | | | |
| NXLZWQ | 420_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 421_BUILD_ADAM3_410_MSG_TABLE | 24 | | | | 5 | | | | 2 | | | | 921984 | | | |
| NXLZWQ | 421_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 4211_BUILD_ADAM3_410_DELE_MSG | 14 | | | | 3 | | | | 2 | | | | 72576 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXLZWQ | 4211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 430_BUILD_ASIF_MSG | 17 | | | | 9 | | | | 3 | | | | 137700 | | | |
| NXLZWQ | 430_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 431_BUILD_ASIF_410_MSG_TABLE | 15 | | | | 4 | | | | 2 | | | | 77760 | | | |
| NXLZWQ | 431_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 700_READ_MOVEMENT_FILE | 9 | | | | 4 | | | | 2 | | | | 1764 | | | |
| NXLZWQ | 700_READ_MOVEMENT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 700_SEND_MESSAGE | 38 | | | | 10 | | | | 2 | | | | 3420000 | | | |
| NXLZWQ | 700_SEND_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 700_READ_AKF | 7 | | | | 3 | | | | 2 | | | | 5103 | | | |
| NXLZWQ | 700_READ_AKF_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 700_READ_ON_DEMAND_FILE | 7 | | | | 4 | | | | 2 | | | | 700 | | | |
| NXLZWQ | 700_READ_ON_DEMAND_FILE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 700_READ_HISTORY_FILE | 11 | | | | 8 | | | | 2 | | | | 7436 | | | |
| NXLZWQ | 700_READ_HISTORY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXLZWQ | 700_EOF_PROCESS | 8 | | | | 4 | | | | 3 | | | | 2048 | | | |
| NXLZWQ | 700_EOF_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXM1WQ | 000_DRIVER | 5 | | | 7 | 2 | | | 3 | 1 | | | 2 | 80 | | | 700 |
| NXM1WQ | 000_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXM1WQ | 100_HOUSEKEEPING | 6 | | | 7 | 2 | | | 3 | 1 | | | 2 | 3456 | | | 5488 |
| NXM1WQ | 100_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXM1WQ | 110_LOAD_ADAM_III_CONUS_TABLE | 9 | | | 10 | 3 | | | 4 | 2 | | | 2 | 2025 | | | 3240 |
| NXM1WQ | 110_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXM1WQ | 200_PROCESS_LB | 7 | | | 8 | 8 | | | 9 | 2 | | | 2 | 4375 | | | 7200 |
| NXM1WQ | 200_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXM1WQ | 300_AKF_SETTER | 8 | | | 9 | 3 | | | 4 | 1 | | | 2 | 648 | | | 1296 |
| NXM1WQ | 300_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXM1WQ | 400_WRAPUP | 2 | | | 3 | 1 | | | 2 | 1 | | | 2 | 0 | | | 0 |
| NXM1WQ | 400_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXM1WQ | 700_READ_LB_MOVEMENT | 4 | | | 5 | 2 | | | 3 | 1 | | | 2 | 36 | | | 80 |
| NXM1WQ | 700_READ_LB_MOVEMENT_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXM1WQ | 700_LB_READ_WRITE | 5 | | | 8 | 3 | | | 5 | 1 | | | 2 | 180 | | | 800 |
| NXM1WQ | 700_LB_READ_WRITE_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXM1WQ | 700_CHECK_ADAM_III_CONUS | 5 | | | 6 | 3 | | | 4 | 1 | | | 2 | 500 | | | 864 |
| NXM1WQ | 700_CHECK_ADAM_III_CONUS_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMAWQ | 000_MA_DRIVER | 4 | | | | 2 | | | | 1 | | | | 144 | | | |
| NXMAWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMAWQ | 100_HSKP | 19 | | | | 2 | | | | 1 | | | | 2657644 | | | |
| NXMAWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMAWQ | 110_SELECT_INPUT | 27 | | | | 27 | | | | 2 | | | | 190512 | | | |
| NXMAWQ | 110_CONTINUE | 2 | | | | 2 | | | | 1 | | | | 2 | | | |
| NXMAWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMAWQ | 200_BUILD_REPORT | 38 | | | | 7 | | | | 3 | | | | 3108248 | | | |
| NXMAWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMAWQ | 700_COMPUTE_VALUES | 2 | | | | 2 | | | | 1 | | | | 72 | | | |
| NXMAWQ | 700_COMPUTE_VALUES_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMAWQ | 710_TYPE_PROC | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXMAWQ | 710_TYPE_PROC_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMAWQ | 711_CNTR_PROC | 6 | | | | 1 | | | | 1 | | | | 7350 | | | |
| NXMAWQ | 711_CNTR_PROC_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMAWQ | 700_READ_SORTED_FILE | 7 | | | | 2 | | | | 1 | | | | 1575 | | | |
| NXMAWQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMAWQ | 700_INCREMENT_COUNTERS | 14 | | | | 7 | | | | 3 | | | | 6174 | | | |
| NXMAWQ | 700_INC_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMAWQ | 701_ORG_LOOSE_SHPMTS | 49 | | | | 1 | | | | 1 | | | | 254016 | | | |
| NXMAWQ | 701_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMAWQ | 702_ORG_PALLETIZED_SHPMTS | 49 | | | | 1 | | | | 1 | | | | 254016 | | | |
| NXMAWQ | 702_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMAWQ | 703_INT_LOOSE_SHPMTS | 49 | | | | 1 | | | | 1 | | | | 254016 | | | |
| NXMAWQ | 703_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMAWQ | 704_INT_PALLETIZED_SHPMTS | 49 | | | | 1 | | | | 1 | | | | 254016 | | | |
| NXMAWQ | 704_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMAWQ | 705_TRM_LOOSE_SHPMTS | 49 | | | | 1 | | | | 1 | | | | 254016 | | | |
| NXMAWQ | 705_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMAWQ | 706_TRM_PALLETIZED_SHPMTS | 49 | | | | 1 | | | | 1 | | | | 254016 | | | |
| NXMAWQ | 706_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXMAWQ | 700_WRITE_REPORT | 90 | | | | 7 | | | | 2 | | | | 10160640 | | | |
| NXMAWQ | 700_WRITE_REPORT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMAWQ | 700_WRITE_TRAILER | 6 | | | | 2 | | | | 1 | | | | 3750 | | | |
| NXMAWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMAWQ | 700_WRITE_HEADERS | 5 | | | | 4 | | | | 1 | | | | 720 | | | |
| NXMAWQ | 700_WRITE_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMBWQ | 000_MB_DRIVER | 13 | | 15 | | 7 | | 8 | | 2 | | 2 | | 4212 | | 11760 | |
| NXMBWQ | 000_STOP_RUN | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| NXMBWQ | 100_HSKP | 24 | | 25 | | 3 | | 4 | | 1 | | 2 | | 9255384 | | 11390625 | |
| NXMBWQ | 100_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| NXMBWQ | 110_SELECT_INPUT | 6 | | 8 | | 3 | | 5 | | 1 | | 2 | | 486 | | 1152 | |
| NXMBWQ | 110_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| NXMBWQ | 200_BUILD_REPORT | 19 | | 22 | | 6 | | 8 | | 2 | | 2 | | 112651 | | 301158 | |
| NXMBWQ | 200_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| NXMBWQ | 210_TOTALS_TABLE_SEARCH | 6 | | 7 | | 4 | | 5 | | 1 | | 2 | | 1350 | | 2268 | |
| NXMBWQ | 210_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| NXMBWQ | 220_UPDATE_TABLE | 60 | | 61 | | 10 | | 11 | | 2 | | 2 | | 150000 | | 184525 | |
| NXMBWQ | 220_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| NXMBWQ | 230_FORMAT_DETAIL_LINE | 23 | | 24 | | 1 | | 2 | | 1 | | 2 | | 5888828 | | 6690816 | |
| NXMBWQ | 230_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| NXMBWQ | 240_FORMAT_DELIM_LINE | | | 24 | | | | 2 | | | | 2 | | | | 6690816 | |
| NXMBWQ | 240_EXIT | | | 1 | | | | 1 | | | | 1 | | | | 0 | |
| NXMBWQ | 300_WRAP_UP | 9 | | 10 | | 2 | | 3 | | 1 | | 2 | | 9216 | | 16000 | |
| NXMBWQ | 300_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| NXMBWQ | 310_UNLOAD_TABLE | 13 | | 14 | | 2 | | 3 | | 1 | | 2 | | 29952 | | 50400 | |
| NXMBWQ | 310_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| NXMBWQ | 320_PORT_RECAP_PROCESS | 10 | | 11 | | 2 | | 3 | | 1 | | 2 | | 7290 | | 14256 | |
| NXMBWQ | 320_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| NXMBWQ | 321_READ_SORTED_TABLE | 10 | | 11 | | 3 | | 4 | | 2 | | 2 | | 4000 | | 6875 | |
| NXMBWQ | 321_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| NXMBWQ | 700_UPDATE_MAC_TOTALS | 30 | | 30 | | 1 | | 1 | | 1 | | 1 | | 25947000 | | 25947000 | |
| NXMBWQ | 700_UPDATE_MAC_TOTALS_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| NXMBWQ | 700_COMPUTE | 10 | | 10 | | 1 | | 1 | | 1 | | 1 | | 144000 | | 144000 | |
| NXMBWQ | 700_COMPUTE_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| NXMBWQ | 700_FORMAT | 21 | | 21 | | 1 | | 1 | | 1 | | 1 | | 5376756 | | 5376756 | |
| NXMBWQ | 700_FORMAT_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| NXMBWQ | 700_WRITE_2 | 17 | | 17 | | 15 | | 15 | | 1 | | 1 | | 39168 | | 39168 | |
| NXMBWQ | 700_WRITE_2_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| NXMBWQ | 700_WRITE_TRAILER | 6 | | 6 | | 2 | | 2 | | 1 | | 1 | | 3750 | | 3750 | |
| NXMBWQ | 700_WRITE_TRAILER_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| NXMBWQ | 700_WRITE_HEADINGS | 8 | | 8 | | 6 | | 6 | | 1 | | 1 | | 1568 | | 1568 | |
| NXMBWQ | 700_HEADINGS_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| NXMDWQ | 000_MD_DRIVER | 7 | | | | 3 | | | | 2 | | | | 1008 | | | |
| NXMDWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMDWQ | 100_HSKP | 23 | | | | 3 | | | | 2 | | | | 3942108 | | | |
| NXMDWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMDWQ | 200_BUILD_REPORT | 17 | | | | 17 | | | | 2 | | | | 22032 | | | |
| NXMDWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMDWQ | 210_PROCESS_RECORD | 27 | | | | 25 | | | | 2 | | | | 33075 | | | |
| NXMDWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMDWQ | 300_WRAPUP | 10 | | | | 6 | | | | 2 | | | | 40960 | | | |
| NXMDWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMDWQ | 310_SUM_PORT | 8 | | | | 3 | | | | 2 | | | | 9800 | | | |
| NXMDWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMDWQ | 320_PORT_TOTAL_TOTAL | 7 | | | | 3 | | | | 2 | | | | 4032 | | | |
| NXMDWQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMDWQ | 330_TOTAL_HOLD_AVG | 4 | | | | 3 | | | | 2 | | | | 400 | | | |
| NXMDWQ | 330_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMDWQ | 340_PRINT_DETAIL | 20 | | | | 3 | | | | 2 | | | | 414720 | | | |
| NXMDWQ | 340_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMDWQ | 700_WRITE_HEADINGS | 8 | | | | 6 | | | | 2 | | | | 800 | | | |
| NXMDWQ | 700_WRITE_HEADINGS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMDWQ | 700_WRITE_TRAILER | 7 | | | | 3 | | | | 2 | | | | 4375 | | | |
| NXMDWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 000_CONTROL | 8 | | | | 3 | | | | 2 | | | | 1800 | | | |
| NXMEWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXMEWQ | 100_HSKP | 24 | | | | 7 | | | | 2 | | | | 1247616 | | | |
| NXMEWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 105_INIT_AGCY_WT_TABLE | 1 | | | | 1 | | | | 1 | | | | 1 | | | |
| NXMEWQ | 105_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 110_LOAD_NAF_TABLE | 17 | | | | 8 | | | | 3 | | | | 88128 | | | |
| NXMEWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 120_LOAD_WEIGHT_TABLE | 11 | | | | 5 | | | | 2 | | | | 4851 | | | |
| NXMEWQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 121_CARD_PROCESS | 13 | | | | 7 | | | | 2 | | | | 26325 | | | |
| NXMEWQ | 121_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 130_TRIGGER_DATA | 19 | | | | 3 | | | | 2 | | | | 1040364 | | | |
| NXMEWQ | 130_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 140_LOAD_MOD_TABLES | 15 | | | | 6 | | | | 2 | | | | 34560 | | | |
| NXMEWQ | 140_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 150_SORT | 9 | | | | 3 | | | | 2 | | | | 20736 | | | |
| NXMEWQ | 150_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 151_SELECT | 40 | | | | 17 | | | | 2 | | | | 8317440 | | | |
| NXMEWQ | 151_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 200_BUILD_REPORT | 30 | | | | 8 | | | | 3 | | | | 768000 | | | |
| NXMEWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 210_INCREMENT_COUNTERS | 21 | | | | 9 | | | | 2 | | | | 254100 | | | |
| NXMEWQ | 210_CONTINUE | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 211_LOOSE_PROCESS | 16 | | | | 6 | | | | 2 | | | | 102400 | | | |
| NXMEWQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 212_PLTZD_PROCESS | 10 | | | | 2 | | | | 2 | | | | 23040 | | | |
| NXMEWQ | 212_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 213_PLTZD_BY_OTHER_AGENCIES | 17 | | | | 9 | | | | 2 | | | | 382500 | | | |
| NXMEWQ | 213_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 214_UNRESTRICTED_PROCESS | 12 | | | | 3 | | | | 2 | | | | 37632 | | | |
| NXMEWQ | 214_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 215_RESTRICTED_PROCESS | 12 | | | | 3 | | | | 2 | | | | 37632 | | | |
| NXMEWQ | 215_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 220_CHANNEL_OUTPUT | 46 | | | | 6 | | | | 2 | | | | 12105774 | | | |
| NXMEWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 230_MFST_STN_OUTPUT | 46 | | | | 7 | | | | 2 | | | | 12105774 | | | |
| NXMEWQ | 230_CONTINUE | 38 | | | | 5 | | | | 2 | | | | 21888 | | | |
| NXMEWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 231_PLT_TRAIN_OUTPUT | 12 | | | | 6 | | | | 2 | | | | 27648 | | | |
| NXMEWQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 2311_PLT_TRAIN_PROC | 21 | | | | 3 | | | | 2 | | | | 354900 | | | |
| NXMEWQ | 2311_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 2312_PLT_TRAIN_TOTAL_PROC | 16 | | | | 3 | | | | 2 | | | | 82944 | | | |
| NXMEWQ | 2312_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 232_LOOSE_CARGO_OUTPUT | 9 | | | | 5 | | | | 2 | | | | 5625 | | | |
| NXMEWQ | 232_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 2321_LOOSE_CARGO_PROC | 25 | | | | 5 | | | | 2 | | | | 270400 | | | |
| NXMEWQ | 2321_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 233_OTHER_AGENCY_OUTPUT | 9 | | | | 5 | | | | 2 | | | | 11664 | | | |
| NXMEWQ | 233_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 2331_OTHER_AGENCY_PROC | 39 | | | | 4 | | | | 2 | | | | 5281536 | | | |
| NXMEWQ | 2331_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 234_NAF_SEARCH | 6 | | | | 4 | | | | 2 | | | | 1176 | | | |
| NXMEWQ | 234_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 300_WRAPUP | 10 | | | | 3 | | | | 2 | | | | 25000 | | | |
| NXMEWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 310_21ST_AIR_FORCE_TOTAL | 58 | | | | 8 | | | | 2 | | | | 36934632 | | | |
| NXMEWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 311_PLT_TRAIN_PROC | 23 | | | | 4 | | | | 2 | | | | 470327 | | | |
| NXMEWQ | 311_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 312_PLT_TRAIN_TOTAL_PROC | 16 | | | | 3 | | | | 2 | | | | 82944 | | | |
| NXMEWQ | 312_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 313_LOOSE_CARGO_PROC | 25 | | | | 5 | | | | 2 | | | | 270400 | | | |
| NXMEWQ | 313_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 320_22ND_AIR_FORCE_TOTAL | 55 | | | | 8 | | | | 2 | | | | 35024220 | | | |
| NXMEWQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXMEWQ | 321_PLT_TRAIN_PROC | 23 | | | | 4 | | | | 2 | | | | 470327 | | | |
| NXMEWQ | 321_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 322_PLT_TRAIN_TOTAL_PROC | 16 | | | | 3 | | | | 2 | | | | 82944 | | | |
| NXMEWQ | 322_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 323_LOOSE_CARGO_PROC | 25 | | | | 5 | | | | 2 | | | | 270400 | | | |
| NXMEWQ | 323_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 330_MAC_HEADQUARTERS_TOTAL | 71 | | | | 8 | | | | 2 | | | | 45213084 | | | |
| NXMEWQ | 330_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 331_PLT_TRAIN_PROC | 23 | | | | 4 | | | | 2 | | | | 470327 | | | |
| NXMEWQ | 331_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 332_PLT_TRAIN_TOTAL_PROC | 16 | | | | 3 | | | | 2 | | | | 82944 | | | |
| NXMEWQ | 332_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 333_LOOSE_CARGO_PROC | 25 | | | | 5 | | | | 2 | | | | 270400 | | | |
| NXMEWQ | 333_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 700_READ_SORTED_FILE | 9 | | | | 4 | | | | 2 | | | | 2916 | | | |
| NXMEWQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 700_WRITE_TRAILER | 7 | | | | 3 | | | | 2 | | | | 34300 | | | |
| NXMEWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMEWQ | 700_WRITE_HEADERS | 17 | | | | 12 | | | | 2 | | | | 244800 | | | |
| NXMEWQ | 700_WRITE_HDRS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMFWQ | 000_MF_DRIVER | 6 | | | | 3 | | | | 2 | | | | 384 | | | |
| NXMFWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMFWQ | 100_HSKP | 22 | | | | 4 | | | | 2 | | | | 3843928 | | | |
| NXMFWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMFWQ | 110_SELECT_INPUT | 18 | | | | 12 | | | | 2 | | | | 27378 | | | |
| NXMFWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMFWQ | 200_BUILD_REPORT | 23 | | | | 4 | | | | 2 | | | | 957168 | | | |
| NXMFWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMFWQ | 210_MF_BREAKS | 41 | | | | 9 | | | | 3 | | | | 2501369 | | | |
| NXMFWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMFWQ | 220_DTL_PROCESS | 31 | | | | 25 | | | | 2 | | | | 24304 | | | |
| NXMFWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMFWQ | 230_GRAND_TOTALS | 34 | | | | 8 | | | | 2 | | | | 982600 | | | |
| NXMFWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMFWQ | 300_WRAPUP | 5 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXMFWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMFWQ | 700_DTL_LINE_OUTPUT | 65 | | | | 9 | | | | 2 | | | | 1834560 | | | |
| NXMFWQ | 700_DTL_LINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMFWQ | 700_CHNL_TOTAL_OUTPUT | 23 | | | | 4 | | | | 2 | | | | 91287 | | | |
| NXMFWQ | 700_CHNL_TOTAL_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMFWQ | 700_SVC_TOTAL_OUTPUT | 107 | | | | 7 | | | | 2 | | | | 4629248 | | | |
| NXMFWQ | 700_SVC_TOTAL_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMFWQ | 700_STN_TOTAL_OUTPUT | 23 | | | | 4 | | | | 2 | | | | 72128 | | | |
| NXMFWQ | 700_STN_TOTAL_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMFWQ | 700_GRND_TOT_BY_SVC_OUTPUT | 13 | | | | 3 | | | | 2 | | | | 22932 | | | |
| NXMFWQ | 700_GRND_TOT_BY_SVC_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMFWQ | 700_GRND_TOTAL_OUTPUT | 13 | | | | 3 | | | | 2 | | | | 11700 | | | |
| NXMFWQ | 700_GRND_TOTAL_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMFWQ | 700_COMPUTE | 3 | | | | 3 | | | | 2 | | | | 1728 | | | |
| NXMFWQ | 700_COMPUTE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMFWQ | 700_COMPUTE_TABLE | 10 | | | | 3 | | | | 2 | | | | 17640 | | | |
| NXMFWQ | 700_COMPUTE_TABLE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMFWQ | 700_WRITE_HEADINGS | 9 | | | | 6 | | | | 2 | | | | 18225 | | | |
| NXMFWQ | 700_HEADINGS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMHWQ | 000_CONTROL | 15 | 16 | | | 2 | 2 | | | 2 | 2 | | | 24000 | 32400 | | |
| NXMHWQ | 000_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 100_HSKP | 18 | 18 | | | 4 | 4 | | | 2 | 2 | | | 209952 | 209952 | | |
| NXMHWQ | 100_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 110_LOAD_NAF_TABLE | 16 | 16 | | | 8 | 8 | | | 3 | 3 | | | 48400 | 48400 | | |
| NXMHWQ | 110_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 120_TRIGGER_DATA | 24 | 24 | | | 3 | 3 | | | 2 | 2 | | | 8871936 | 8871936 | | |
| NXMHWQ | 120_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 130_SORT | 8 | 8 | | | 3 | 3 | | | 2 | 2 | | | 12800 | 12800 | | |
| NXMHWQ | 130_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 131_SELECT_INPUT | 27 | 30 | | | 13 | 14 | | | 2 | 2 | | | 456300 | 711480 | | |
| NXMHWQ | 131_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXMHWQ | 200_AIR | 21 | 21 | | | 8 | 8 | | | 2 | 2 | | | 170100 | 170100 | | |
| NXMHWQ | 200_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 300_TRUCK | 24 | 24 | | | 9 | 9 | | | 2 | 2 | | | 653400 | 653400 | | |
| NXMHWQ | 300_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 400_MH_OUTPUT | 13 | 13 | | | 2 | 2 | | | 2 | 2 | | | 269568 | 269568 | | |
| NXMHWQ | 400_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 410_AIR_PROCESS | 11 | 11 | | | 5 | 5 | | | 2 | 2 | | | 45056 | 45056 | | |
| NXMHWQ | 410_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 411_STATION_PROCESS | 47 | 47 | | | 9 | 9 | | | 2 | 2 | | | 31142012 | 31142012 | | |
| NXMHWQ | 411_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 411A_STATION_COMPUTATION | 42 | 42 | | | 7 | 7 | | | 2 | 2 | | | 1959552 | 1959552 | | |
| NXMHWQ | 411A_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 420_AIR_TOTALS | 92 | 117 | | | 13 | 17 | | | 2 | 2 | | | 248349308 | 635726637 | | |
| NXMHWQ | 420_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 430_TRK_PROCESS | 12 | 12 | | | 5 | 5 | | | 2 | 2 | | | 76800 | 76800 | | |
| NXMHWQ | 430_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 431_STATION_PROCESS | 47 | 47 | | | 9 | 9 | | | 2 | 2 | | | 31142012 | 31142012 | | |
| NXMHWQ | 431_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 431A_STATION_COMPUTATION | 21 | 21 | | | 4 | 4 | | | 2 | 2 | | | 244944 | 244944 | | |
| NXMHWQ | 431A_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 440_TRK_TOTALS | 58 | 81 | | | 9 | 11 | | | 2 | 2 | | | 51795450 | 170772624 | | |
| NXMHWQ | 440_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 700_NAF_SEARCH | 6 | 6 | | | 4 | 4 | | | 2 | 2 | | | 1536 | 1536 | | |
| NXMHWQ | 700_NAF_SEARCH_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 700_AIR_HEADINGS | 9 | 9 | | | 7 | 7 | | | 2 | 2 | | | 1764 | 1764 | | |
| NXMHWQ | 700_AIR_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 700_TRK_HEADINGS | 9 | 9 | | | 7 | 7 | | | 2 | 2 | | | 1764 | 1764 | | |
| NXMHWQ | 700_TRK_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 700_AIR_TOTAL_HDRS | 9 | 9 | | | 7 | 7 | | | 2 | 2 | | | 1296 | 1296 | | |
| NXMHWQ | 700_AT_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 700_TRK_TOTAL_HDRS | 9 | 9 | | | 7 | 7 | | | 2 | 2 | | | 1296 | 1296 | | |
| NXMHWQ | 700_TT_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 710_MODULE_ID_PROCESS | 19 | 19 | | | 9 | 9 | | | 2 | 2 | | | 822016 | 822016 | | |
| NXMHWQ | 710_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 711_MODULE_ID_COMPUTATION | 12 | 12 | | | 3 | 3 | | | 2 | 2 | | | 78732 | 78732 | | |
| NXMHWQ | 711_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 720_MFST_REF_PROCESS | 22 | 22 | | | 7 | 7 | | | 2 | 2 | | | 1627648 | 1627648 | | |
| NXMHWQ | 720_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 721_MANIF_REF_COMPUTATION | 17 | 17 | | | 6 | 6 | | | 2 | 2 | | | 108800 | 108800 | | |
| NXMHWQ | 721_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 700_READ_AIR | 4 | 4 | | | 3 | 3 | | | 2 | 2 | | | 144 | 144 | | |
| NXMHWQ | 700_READ_AIR_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 700_READ_TRUCK | 4 | 4 | | | 3 | 3 | | | 2 | 2 | | | 144 | 144 | | |
| NXMHWQ | 700_READ_TRUCK_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMHWQ | 700_DETERMINE_TYPE | 8 | 8 | | | 8 | 8 | | | 2 | 2 | | | 128 | 128 | | |
| NXMHWQ | 700_TYPE_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXMIWQ | 000_MI_CONTROL | 5 | | | | 2 | | | | 1 | | | | 180 | | | |
| NXMIWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMIWQ | 100_HSKP | 33 | | | | 5 | | | | 1 | | | | 26492928 | | | |
| NXMIWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMIWQ | 110_SELECT_INPUT_1 | 22 | | | | 14 | | | | 2 | | | | 228888 | | | |
| NXMIWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMIWQ | 120_SELECT_INPUT_2 | 22 | | | | 14 | | | | 2 | | | | 228888 | | | |
| NXMIWQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMIWQ | 130_SELECT_INPUT_3 | 21 | | | | 14 | | | | 2 | | | | 134400 | | | |
| NXMIWQ | 130_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMIWQ | 200_BUILD_REPORT | 21 | | | | 6 | | | | 2 | | | | 365904 | | | |
| NXMIWQ | 200_CONTINUE | 2 | | | | 2 | | | | 2 | | | | 2 | | | |
| NXMIWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMIWQ | 210_DELIVERY_TERMS_PROC | 23 | | | | 4 | | | | 2 | | | | 1154048 | | | |
| NXMIWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMIWQ | 220_SERVICE_BRK_PROC | 34 | | | | 9 | | | | 2 | | | | 574600 | | | |
| NXMIWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMIWQ | 230_COUNTRY_BRK_PROC | 19 | | | | 5 | | | | 2 | | | | 147136 | | | |
| NXMIWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMIWQ | 240_PROCESS | 9 | | | | 4 | | | | 2 | | | | 11025 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXMIWQ | 240_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMIWQ | 300_WRAP_UP | 61 | | | | 10 | | | | 2 | | | | 4954725 | | | |
| NXMIWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMIWQ | 700_WRITE_HEADINGS | 8 | | | | 6 | | | | 1 | | | | 2592 | | | |
| NXMIWQ | 700_HEADINGS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMIWQ | 700_WRITE_HEADINGS_F | 6 | | | | 4 | | | | 1 | | | | 216 | | | |
| NXMIWQ | 700_HEADINGS_F_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMIWQ | 700_WRITE_TRAILER | 6 | | | | 2 | | | | 1 | | | | 9600 | | | |
| NXMIWQ | 700_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMIWQ | 700_COUNTRY_SRCH | 5 | | | | 3 | | | | 1 | | | | 2420 | | | |
| NXMIWQ | 700_COUNTRY_SRCH_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMIWQ | 700_SERVICE_SRCH | 5 | | | | 3 | | | | 1 | | | | 980 | | | |
| NXMIWQ | 700_SERVICE_SRCH_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 000_MJ_CONTROL | 11 | | | | 4 | | | | 2 | | | | 13475 | | | |
| NXMJWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 100_HSKP | 12 | | | | 3 | | | | 2 | | | | 21168 | | | |
| NXMJWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 110_CURRENT_DATA | 15 | | | | 4 | | | | 2 | | | | 47040 | | | |
| NXMJWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 120_BUILD_CHNL_TABLE | 5 | | | | 2 | | | | 1 | | | | 720 | | | |
| NXMJWQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 130_SORT | 9 | | | | 2 | | | | 1 | | | | 11664 | | | |
| NXMJWQ | 130_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 131_SELECT | 42 | | | | 18 | | | | 2 | | | | 1680000 | | | |
| NXMJWQ | 131_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 1311_CHNL_SEARCH | 5 | | | | 4 | | | | 1 | | | | 980 | | | |
| NXMJWQ | 1311_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 200_BUILD_REPORT | 30 | | | | 6 | | | | 2 | | | | 1285470 | | | |
| NXMJWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 210_CHNL_OUTPUT | 3 | | | | 2 | | | | 1 | | | | 675 | | | |
| NXMJWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 220_COMPUTE_TOTALS | 15 | | | | 3 | | | | 1 | | | | 365040 | | | |
| NXMJWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 230_WRITE_HEADERS | 10 | | | | 7 | | | | 1 | | | | 4410 | | | |
| NXMJWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 2301_FILL_HDRS_TRLR | 17 | | | | 1 | | | | 1 | | | | 2506752 | | | |
| NXMJWQ | 2301_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 240_WRITE_TOTAL_HEADERS | 10 | | | | 7 | | | | 1 | | | | 4410 | | | |
| NXMJWQ | 240_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 250_AGE_PROC | 41 | | | | 15 | | | | 2 | | | | 850176 | | | |
| NXMJWQ | 250_AGE_PROC_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 2501_CNTR_OUTPUT | 6 | | | | 1 | | | | 1 | | | | 7350 | | | |
| NXMJWQ | 2501_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 260_PROCESS | 268 | | | | 37 | | | | 2 | | | | 4669632 | | | |
| NXMJWQ | 260_PROCESS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 300_WRAP_UP | 24 | | | | 1 | | | | 1 | | | | 1161600 | | | |
| NXMJWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 700_APOE_WORK | 7 | | | | 1 | | | | 1 | | | | 12348 | | | |
| NXMJWQ | 700_APOE_WORK_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 700_APOE_AVERAGE_TIMES | 6 | | | | 1 | | | | 1 | | | | 3750 | | | |
| NXMJWQ | 700_APOE_AVERAGE_TIMES_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMJWQ | 700_WRITE_TRAILER | 6 | | | | 2 | | | | 1 | | | | 7350 | | | |
| NXMJWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMPWQ | 000_MP_DRIVER | 7 | | | 9 | 2 | | | 3 | 2 | | | 2 | 112 | | | 900 |
| NXMPWQ | 000_STOP_RUN | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMPWQ | 100_HSKP | 46 | | | 48 | 9 | | | 11 | 1 | | | 2 | 29292984 | | | 32039472 |
| NXMPWQ | 100_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMPWQ | 105_LOAD_ADAM_III_TABLE | 6 | | | 6 | 2 | | | 2 | 1 | | | 1 | 486 | | | 486 |
| NXMPWQ | 105_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMPWQ | 110_PZ4_ORIGINATING_MONTH_FILE | 26 | | | 27 | 16 | | | 17 | 2 | | | 2 | 531674 | | | 640332 |
| NXMPWQ | 110_CONTINUE | 23 | | | 23 | 8 | | | 8 | 2 | | | 2 | 1868175 | | | 1868175 |
| NXMPWQ | 110_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMPWQ | 120_PZ4_TERMINATING_MONTH_FILE | 26 | | | 27 | 14 | | | 15 | 2 | | | 2 | 380666 | | | 470448 |
| NXMPWQ | 120_CONTINUE | 24 | | | 27 | 8 | | | 10 | 2 | | | 2 | 2217984 | | | 3158028 |
| NXMPWQ | 120_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMPWQ | 130_PZ3_OLD_MONTH_FILE | 38 | | | 41 | 21 | | | 23 | 3 | | | 3 | 2318342 | | | 2771600 |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXMPWQ | 130_CONTINUE | 20 | | | 20 | 4 | | | 4 | 2 | | | 2 | 1458000 | | | 1458000 |
| NXMPWQ | 130_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMPWQ | 200_CREATE_FILE_DRIVER | 5 | | | 6 | 2 | | | 3 | 1 | | | 2 | 1620 | | | 2646 |
| NXMPWQ | 200_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMPWQ | 210_CREATE_MINI_FILE_FOR_RPT | 18 | | | 19 | 17 | | | 18 | 4 | | | 4 | 127008 | | | 157339 |
| NXMPWQ | 210_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMPWQ | 211_ORIG_FOR_MINI_REC | 31 | | | 34 | 24 | | | 27 | 2 | | | 2 | 1290096 | | | 1586304 |
| NXMPWQ | 211_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMPWQ | 212_COMPUTE_PHTS | 54 | | | 55 | 7 | | | 8 | 2 | | | 2 | 15283296 | | | 17248000 |
| NXMPWQ | 212_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMPWQ | 300_WRAPUP | 45 | | | 46 | 1 | | | 2 | 1 | | | 2 | 0 | | | 0 |
| NXMPWQ | 300_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMPWQ | 700_READ | 6 | | | 7 | 2 | | | 3 | 1 | | | 2 | 216 | | | 448 |
| NXMPWQ | 700_READ_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMPWQ | 700_ADAM_III_CHECK | 5 | | | 6 | 3 | | | 4 | 1 | | | 2 | 980 | | | 1536 |
| NXMPWQ | 700_ADAM_III_CHECK_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMPWQ | 700_COMPUTE_HOURS | 16 | | | 17 | 6 | | | 7 | 2 | | | 2 | 186624 | | | 220932 |
| NXMPWQ | 700_COMPUTE_HOURS_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMPWQ | 700_WRITE_OVER_10_DAY | 8 | | | 9 | 2 | | | 3 | 1 | | | 2 | 14112 | | | 21609 |
| NXMPWQ | 700_WRITE_OVER_10_DAY_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMPWQ | 700_READ_MOVE | 13 | | | 15 | 7 | | | 9 | 2 | | | 2 | 25168 | | | 34560 |
| NXMPWQ | 700_READ_MOVE_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMPWQ | 700_ERROR_RTN | 4 | | | 5 | 1 | | | 2 | 1 | | | 2 | 196 | | | 320 |
| NXMPWQ | 700_ERROR_RTN_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| NXMQWQ | 000_MQ_DRIVER | 6 | | | | 3 | | | | 2 | | | | 384 | | | |
| NXMQWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMQWQ | 100_HSKP | 24 | | | | 4 | | | | 2 | | | | 10967424 | | | |
| NXMQWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMQWQ | 110_CREATE_SORTED_FILE | 6 | | | | 4 | | | | 2 | | | | 384 | | | |
| NXMQWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMQWQ | 200_PROCESS | 12 | | | | 6 | | | | 2 | | | | 12288 | | | |
| NXMQWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMQWQ | 300_WRAPUP | 3 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXMQWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMQWQ | 700_READ_MOVE | 28 | | | | 9 | | | | 2 | | | | 2621808 | | | |
| NXMQWQ | 700_READ_MOVE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMQWQ | 700_HEADERS | 12 | | | | 9 | | | | 2 | | | | 8748 | | | |
| NXMQWQ | 700_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMQWQ | 700_TRAILER | 10 | | | | 4 | | | | 2 | | | | 5760 | | | |
| NXMQWQ | 700_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMRWQ | 000_MR_DRIVER | 12 | | | | 6 | | | | 2 | | | | 15552 | | | |
| NXMRWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMRWQ | 100_HSKP | 19 | | | | 4 | | | | 2 | | | | 2743600 | | | |
| NXMRWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMRWQ | 110_SELECT_TYPE_INPUT | 77 | | | | 38 | | | | 2 | | | | 14907200 | | | |
| NXMRWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMRWQ | 200_SELECT_INPUT | 51 | | | | 45 | | | | 3 | | | | 734400 | | | |
| NXMRWQ | 200_CONTINUE | 3 | | | | 3 | | | | 2 | | | | 27 | | | |
| NXMRWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMRWQ | 300_AREA_PRI_PROCESS | 10 | | | | 5 | | | | 2 | | | | 9000 | | | |
| NXMRWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMRWQ | 310_PROCESS_TOTALS | 28 | | | | 7 | | | | 3 | | | | 664048 | | | |
| NXMRWQ | 310_CONTINUE | 8 | | | | 4 | | | | 2 | | | | 2048 | | | |
| NXMRWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMRWQ | 400_WRAPUP | 3 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXMRWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMRWQ | 700_CHANNEL_ACCUMULATION | 37 | | | | 9 | | | | 3 | | | | 37594368 | | | |
| NXMRWQ | 700_CHANNEL_ACCUMULATION_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMRWQ | 700_CREATE_CHANNEL_REPORT | 99 | | | | 25 | | | | 2 | | | | 172497600 | | | |
| NXMRWQ | 700_CREATE_CHANNEL_REPORT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMRWQ | 700_CREATE_APOE_REPORT | 72 | | | | 17 | | | | 2 | | | | 28576800 | | | |
| NXMRWQ | 700_CREATE_APOE_REPORT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMRWQ | 700_CREATE_FINAL_REPORT | 70 | | | | 17 | | | | 2 | | | | 30123520 | | | |
| NXMRWQ | 700_CREATE_FINAL_REPORT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMRWQ | 700_TRAILER | 9 | | | | 3 | | | | 2 | | | | 11025 | | | |
| NXMRWQ | 700_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXMRWQ | 700_ERROR_RTN | 4 | | | | 2 | | | | 2 | | | | 196 | | | |
| NXMRWQ | 700_ERROR_RTN_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 000_MS_DRIVER | 6 | | | | 3 | | | | 2 | | | | 384 | | | |
| NXMSWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 100_HSKP | 37 | | | | 4 | | | | 2 | | | | 46247188 | | | |
| NXMSWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 110_READ_WRITE_SORT | 6 | | | | 4 | | | | 2 | | | | 384 | | | |
| NXMSWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 200_PROCESS | 30 | | | | 18 | | | | 2 | | | | 476280 | | | |
| NXMSWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 210_CONUS_OUT | 31 | | | | 13 | | | | 4 | | | | 1891279 | | | |
| NXMSWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 220_RETROGRADE | 31 | | | | 13 | | | | 4 | | | | 1891279 | | | |
| NXMSWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 230_INTRA_A | 31 | | | | 13 | | | | 4 | | | | 1891279 | | | |
| NXMSWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 240_INTRA_P | 31 | | | | 13 | | | | 4 | | | | 1891279 | | | |
| NXMSWQ | 240_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 300_WRAPUP | 3 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXMSWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 700_READ | 5 | | | | 3 | | | | 2 | | | | 320 | | | |
| NXMSWQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 700_ACCUMULATION | 177 | | | | 75 | | | | 2 | | | | 1901915268 | | | |
| NXMSWQ | 700_ACCUMULATION_CONTINUE_1 | 172 | | | | 73 | | | | 2 | | | | 549874368 | | | |
| NXMSWQ | 700_ACCUMULATION_CONTINUE_2 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| NXMSWQ | 700_ACCUMULATION_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| NXMSWQ | 700_WRITE_PRIORITY_PAGE | 47 | | | | 24 | | | | 2 | | | | 23293952 | | | |
| NXMSWQ | 700_WRITE_PRIORITY_PAGE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 700_WRITE_CHANNEL_PAGE | 43 | | | | 21 | | | | 2 | | | | 14266368 | | | |
| NXMSWQ | 700_WRITE_CHANNEL_PAGE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 700_WRITE_AREA_PAGE | 50 | | | | 26 | | | | 2 | | | | 10351250 | | | |
| NXMSWQ | 700_WRITE_AREA_PAGE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 700_WRITE_FINAL_REPORT | 40 | | | | 20 | | | | 3 | | | | 1713960 | | | |
| NXMSWQ | 700_WRITE_FINAL_REPORT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 700_COMPUTE_PCT | 459 | | | | 40 | | | | 2 | | | | 2062929600 | | | |
| NXMSWQ | 700_COMPUTE_PCT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 700_COMPUTE_PCT_A | 326 | | | | 21 | | | | 2 | | | | 626244696 | | | |
| NXMSWQ | 700_COMPUTE_PCT_A_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 700_COMPUTE_PCT_C | 326 | | | | 21 | | | | 2 | | | | 626244696 | | | |
| NXMSWQ | 700_COMPUTE_PCT_C_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 700_M_DAY | 10 | | | | 2 | | | | 2 | | | | 23040 | | | |
| NXMSWQ | 700_M_DAY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 700_M_DAY_A | 10 | | | | 2 | | | | 2 | | | | 19360 | | | |
| NXMSWQ | 700_M_DAY_A_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 700_M_DAY_C | 10 | | | | 2 | | | | 2 | | | | 19360 | | | |
| NXMSWQ | 700_M_DAY_C_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMSWQ | 700_TRAILERS | 9 | | | | 4 | | | | 2 | | | | 26244 | | | |
| NXMSWQ | 700_TRAILERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMTWQ | 000_MT_DRIVER | 8 | | | | 3 | | | | 2 | | | | 512 | | | |
| NXMTWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMTWQ | 100_HSKP | 81 | | | | 25 | | | | 3 | | | | 44475561 | | | |
| NXMTWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMTWQ | 105_LOAD_CHNL_TABLE | 7 | | | | 3 | | | | 2 | | | | 1008 | | | |
| NXMTWQ | 105_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMTWQ | 110_ORIGINATING_MONTH_PZ4 | 92 | | | | 44 | | | | 3 | | | | 286276032 | | | |
| NXMTWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMTWQ | 120_TERMINATING_MONTH_PZ4 | 93 | | | | 42 | | | | 3 | | | | 266246352 | | | |
| NXMTWQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMTWQ | 130_PZ3_OLD_MONTH_FILE | 101 | | | | 45 | | | | 3 | | | | 373879376 | | | |
| NXMTWQ | 130_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMTWQ | 200_CREATE_FILE_DRIVER | 7 | | | | 3 | | | | 2 | | | | 16128 | | | |
| NXMTWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMTWQ | 210_CREATE_MINI_FILE_FOR_RPT | 17 | | | | 18 | | | | 4 | | | | 140777 | | | |
| NXMTWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMTWQ | 211_ORIG_FOR_MINI_REC | 28 | | | | 29 | | | | 2 | | | | 28672 | | | |
| NXMTWQ | 211_CONTINUE | 7 | | | | 1 | | | | 1 | | | | 5488 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXMTWQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMTWQ | 212_COMPUTE_PHTS | 60 | | | | 8 | | | | 2 | | | | 36504000 | | | |
| NXMTWQ | 212_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMTWQ | 300_WRAPUP | 59 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXMTWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMTWQ | 700_READ | 7 | | | | 3 | | | | 2 | | | | 448 | | | |
| NXMTWQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMTWQ | 700_CHNL_CHECK | 8 | | | | 4 | | | | 2 | | | | 12800 | | | |
| NXMTWQ | 700_CHNL_CHECK_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMTWQ | 700_COMPUTE_HOURS | 26 | | | | 10 | | | | 4 | | | | 1146600 | | | |
| NXMTWQ | 700_COMPUTE_HOURS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXMTWQ | 700_READ_MOVE | 14 | | | | 8 | | | | 2 | | | | 32256 | | | |
| NXMTWQ | 700_READ_MOVE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXP4WQ | 000_DRIVER | 4 | | | | 2 | | | | 1 | | | | 36 | | | |
| NXP4WQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXP4WQ | 100_HSKP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXP4WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXP4WQ | 200_READ_WRITE | 7 | | | | 3 | | | | 1 | | | | 1008 | | | |
| NXP4WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXP4WQ | 300_WRAPUP | 3 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXP4WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPAWQ | 000_MAIN_DRIVER | 8 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPAWQ | 000_MAIN_LOOP | 10 | | | | 6 | | | | 2 | | | | 2250 | | | |
| NXPAWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPAWQ | 100_I_CALL | 4 | | | | 1 | | | | 1 | | | | 256 | | | |
| NXPAWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPAWQ | 200_TYPE_INPUT_SELECT | 13 | | | | 6 | | | | 2 | | | | 1872 | | | |
| NXPAWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPAWQ | 210_MINI_INPUT | 7 | | | | 5 | | | | 2 | | | | 252 | | | |
| NXPAWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPAWQ | 220_PPAL_INPUT | 7 | | | | 5 | | | | 2 | | | | 448 | | | |
| NXPAWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPAWQ | 230_CRT_INPUT | 9 | | | | 4 | | | | 2 | | | | 324 | | | |
| NXPAWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPAWQ | 240_CARDXA_INPUT | 16 | | | | 9 | | | | 2 | | | | 10000 | | | |
| NXPAWQ | 240_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPAWQ | 300_NULL_CALL | 5 | | | | 3 | | | | 2 | | | | 245 | | | |
| NXPAWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPAWQ | 400_PROGRAM_SELECT | 27 | | | | 5 | | | | 2 | | | | 139968 | | | |
| NXPAWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPAWQ | 900_WRAPUP | 1 | | | | 1 | | | | 1 | | | | 16 | | | |
| NXPAWQ | 900_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPBWQ | 000_DRIVER | 3 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPBWQ | 000_STOP | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPBWQ | 100_HSKP | 7 | | | | 3 | | | | 1 | | | | 4032 | | | |
| NXPBWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPBWQ | 110_READ_STATION_TO_SORT | 6 | | | | 3 | | | | 1 | | | | 216 | | | |
| NXPBWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPBWQ | 120_WRITE_REC_BACK | 6 | | | | 3 | | | | 1 | | | | 216 | | | |
| NXPBWQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPBWQ | 200_WRAPUP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPBWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPCWQ | 000_DRIVER | 7 | | | | 2 | | | | 2 | | | | 175 | | | |
| NXPCWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPCWQ | 100_INTITIALIZE | 6 | | | | 1 | | | | 1 | | | | 1536 | | | |
| NXPCWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPCWQ | 200_SET_ARGS | 4 | | | | 1 | | | | 1 | | | | 64 | | | |
| NXPCWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPCWQ | 300_CALL_RCOMC | 2 | | | | 1 | | | | 1 | | | | 50 | | | |
| NXPCWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPCWQ | 400_CHECK_ERROR | 9 | | | | 3 | | | | 2 | | | | 324 | | | |
| NXPCWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPCWQ | 410_CHECK_STATIS | 11 | | | | 8 | | | | 2 | | | | 396 | | | |
| NXPCWQ | 410_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPCWQ | 500_BOJ_CALL | 8 | | | | 2 | | | | 2 | | | | 18432 | | | |
| NXPCWQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXPDWQ | 000_MAIN_DRIVER | 11 | | | | 6 | | | | 2 | | | | 1584 | | | |
| NXPDWQ | 000_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 100_HSKP | 16 | | | | 3 | | | | 2 | | | | 112896 | | | |
| NXPDWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 200_PROCESS_MESSAGE | 29 | | | | 24 | | | | 2 | | | | 1254656 | | | |
| NXPDWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 210_MESSAGE_HSKP | 23 | | | | 7 | | | | 2 | | | | 2070000 | | | |
| NXPDWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 220_PROCESS_HISTORICAL_PV_DATA | 5 | | | | 7 | | | | 2 | | | | 720 | | | |
| NXPDWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 221_GETT_CHECK_PROCESS_HEADER | 42 | | | | 8 | | | | 2 | | | | 40832232 | | | |
| NXPDWQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 222_GETT_CHECK_PROCESS_DETAIL | 61 | | | | 17 | | | | 2 | | | | 211489684 | | | |
| NXPDWQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 230_PROCESS_HISTORICAL_PY_DATA | 5 | | | | 7 | | | | 2 | | | | 720 | | | |
| NXPDWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 231_GETT_CHECK_PROCESS_HEADER | 42 | | | | 8 | | | | 2 | | | | 40832232 | | | |
| NXPDWQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 232_GETT_CHECK_PROCESS_DETAIL | 61 | | | | 17 | | | | 2 | | | | 211489684 | | | |
| NXPDWQ | 232_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 240_PROCESS_DAILY_DATA | 4 | | | | 4 | | | | 2 | | | | 256 | | | |
| NXPDWQ | 240_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 241_RETRIEVE_K1_TCMD_RECORD | 40 | | | | 19 | | | | 2 | | | | 7225000 | | | |
| NXPDWQ | 241_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 242_RETRIEVE_K1_TAA_RECORD | 72 | | | | 12 | | | | 2 | | | | 755827200 | | | |
| NXPDWQ | 242_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 250_PROCESS_ON_HAND_DATA | 105 | | | | 40 | | | | 2 | | | | 1585604580 | | | |
| NXPDWQ | 250_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 260_PROCESS_HISTORICAL_CV_DATA | 5 | | | | 7 | | | | 2 | | | | 720 | | | |
| NXPDWQ | 260_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 261_GETT_CHECK_PROCESS_HEADER | 42 | | | | 8 | | | | 2 | | | | 40832232 | | | |
| NXPDWQ | 261_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 262_GETT_CHECK_PROCESS_DETAIL | 61 | | | | 17 | | | | 2 | | | | 211489684 | | | |
| NXPDWQ | 262_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 270_PROCESS_DAILY_COMM_DATA | 4 | | | | 4 | | | | 2 | | | | 256 | | | |
| NXPDWQ | 270_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 271_RETRIEVE_M1_TCMD_RECORD | 40 | | | | 19 | | | | 2 | | | | 7225000 | | | |
| NXPDWQ | 271_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 272_RETRIEVE_M1_TAA_RECORD | 72 | | | | 12 | | | | 2 | | | | 755827200 | | | |
| NXPDWQ | 272_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 280_FILE_LOCKED_ROUTINE | 14 | | | | 5 | | | | 2 | | | | 8064 | | | |
| NXPDWQ | 280_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 300_WRAPUP | 8 | | | | 2 | | | | 2 | | | | 14112 | | | |
| NXPDWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 700_SEND_MESSAGE | 36 | | | | 6 | | | | 2 | | | | 4769856 | | | |
| NXPDWQ | 700_SEND_MESSAGE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 700_MULTI_PART_ROUTINE | 8 | | | | 1 | | | | 1 | | | | 51200 | | | |
| NXPDWQ | 700_MULTI_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 700_ERROR_DUMP | 5 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPDWQ | 700_ERROR_DUMP_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 000_MAIN_DRIVER | 12 | | | | 6 | | | | 2 | | | | 1728 | | | |
| NXPEWQ | 000_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 100_HSKP | 24 | | | | 4 | | | | 2 | | | | 470400 | | | |
| NXPEWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 200_PROCESS_MESSAGE | 30 | | | | 23 | | | | 2 | | | | 1140750 | | | |
| NXPEWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 210_MESSAGE_HSKP | 25 | | | | 7 | | | | 2 | | | | 2640625 | | | |
| NXPEWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 220_PROCESS_HISTORICAL_PV_DATA | 6 | | | | 7 | | | | 2 | | | | 864 | | | |
| NXPEWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 221_GETT_CHECK_PROCESS_HEADER | 44 | | | | 8 | | | | 2 | | | | 42776624 | | | |
| NXPEWQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 222_GETT_CHECK_PROCESS_DETAIL | 64 | | | | 17 | | | | 2 | | | | 221890816 | | | |
| NXPEWQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 230_PROCESS_HISTORICAL_PY_DATA | 6 | | | | 7 | | | | 2 | | | | 864 | | | |
| NXPEWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXPEWQ | 231_GETT_CHECK_PROCESS_HEADER | 44 | | | | 8 | | | | 2 | | | | 42776624 | | | |
| NXPEWQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 232_GETT_CHECK_PROCESS_DETAIL | 64 | | | | 17 | | | | 2 | | | | 221890816 | | | |
| NXPEWQ | 232_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 240_PROCESS_DAILY_DATA | 5 | | | | 4 | | | | 2 | | | | 320 | | | |
| NXPEWQ | 240_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 241_RETRIEVE_K1_TCMD_RECORD | 45 | | | | 19 | | | | 2 | | | | 9856080 | | | |
| NXPEWQ | 241_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 242_RETRIEVE_K1_TAA_RECORD | 75 | | | | 12 | | | | 2 | | | | 844201875 | | | |
| NXPEWQ | 242_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 250_PROCESS_ON_HAND_DATA | 108 | | | | 40 | | | | 2 | | | | 1687630572 | | | |
| NXPEWQ | 250_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 260_PROCESS_HISTORICAL_CV_DATA | 6 | | | | 7 | | | | 2 | | | | 864 | | | |
| NXPEWQ | 260_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 261_GETT_CHECK_PROCESS_HEADER | 44 | | | | 8 | | | | 2 | | | | 42776624 | | | |
| NXPEWQ | 261_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 262_GETT_CHECK_PROCESS_DETAIL | 64 | | | | 17 | | | | 3 | | | | 221890816 | | | |
| NXPEWQ | 262_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 270_PROCESS_DAILY_COMM_DATA | 5 | | | | 4 | | | | 2 | | | | 320 | | | |
| NXPEWQ | 270_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 271_RETRIEVE_M1_TCMD_RECORD | 45 | | | | 19 | | | | 2 | | | | 9856080 | | | |
| NXPEWQ | 271_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 272_RETRIEVE_M1_TAA_RECORD | 75 | | | | 12 | | | | 2 | | | | 844201875 | | | |
| NXPEWQ | 272_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 280_FILE_LOCKED_ROUTINE | 15 | | | | 5 | | | | 2 | | | | 8640 | | | |
| NXPEWQ | 280_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 300_WRAPUP | 24 | | | | 3 | | | | 2 | | | | 743424 | | | |
| NXPEWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 700_SEND_MESSAGE | 50 | | | | 5 | | | | 2 | | | | 7144200 | | | |
| NXPEWQ | 700_SEND_MESSAGE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 700_MULTI_PART_ROUTINE | 9 | | | | 1 | | | | 1 | | | | 57600 | | | |
| NXPEWQ | 700_MULTI_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 700_ERROR_DUMP | 6 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPEWQ | 700_ERROR_DUMP_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPFWQ | 000_MAIN_DRIVER | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPFWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPFWQ | 100_SEND_CRT_SCREEN | 14 | | | | 2 | | | | 2 | | | | 72576 | | | |
| NXPFWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPHWQ | 000_DRIVER | 3 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPHWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPHWQ | 100_SET_O_CALL | 8 | | | | 2 | | | | 2 | | | | 10368 | | | |
| NXPHWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPHWQ | 110_SIGN_OFF_CALLS | 18 | | | | 6 | | | | 2 | | | | 56448 | | | |
| NXPHWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPHWQ | 200_CHECK_ERROR | 7 | | | | 3 | | | | 2 | | | | 112 | | | |
| NXPHWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPJWQ | 000_DRIVER | 4 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPJWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPJWQ | 100_INTITIALIZE | 11 | | | | 1 | | | | 1 | | | | 45056 | | | |
| NXPJWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPJWQ | 200_CONTROL | 53 | | | | 15 | | | | 2 | | | | 2270997 | | | |
| NXPJWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPJWQ | 210_STARS_DISPLAY | 10 | | | | 5 | | | | 2 | | | | 0 | | | |
| NXPJWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPJWQ | 220_CALL_T_PROGRAMMER | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPJWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPJWQ | 300_RKILL | 1 | | | | 1 | | | | 1 | | | | 25 | | | |
| NXPJWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPKWQ | 000_EOD_CONTROL | 8 | | | | 5 | | | | 2 | | | | 800 | | | |
| NXPKWQ | 000_CONTINUE | 6 | | | | 3 | | | | 2 | | | | 4704 | | | |
| NXPKWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPKWQ | 100_EOD_HSKP | 15 | | | | 3 | | | | 2 | | | | 121500 | | | |
| NXPKWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPKWQ | 200_PROCESS_ADAM3_RECS | 36 | | | | 9 | | | | 2 | | | | 2073600 | | | |
| NXPKWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPKWQ | 300_PROCESS_NON_ADAM3_RECS | 34 | | | | 9 | | | | 2 | | | | 1645600 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXPKWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPKWQ | 400_WRITE_SEQ_FROM_SORT | 3 | | | | 2 | | | | 1 | | | | 48 | | | |
| NXPKWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPKWQ | 410_READ_WRITE | 19 | | | | 12 | | | | 2 | | | | 6156 | | | |
| NXPKWQ | 410_CONTINUE | 2 | | | | 2 | | | | 1 | | | | 8 | | | |
| NXPKWQ | 410_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPKWQ | 500_WRAPUP | 36 | | | | 4 | | | | 2 | | | | 2509056 | | | |
| NXPKWQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPKWQ | 700_ERROR_DISPLAY | 7 | | | | 1 | | | | 1 | | | | 1008 | | | |
| NXPKWQ | 700_ERROR_DISPLAY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPLWQ | 000_DRIVER | 23 | | | | 9 | | | | 2 | | | | 476928 | | | |
| NXPLWQ | 000_STOP | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPLWQ | 100_HSKP | 51 | | | | 6 | | | | 2 | | | | 11897739 | | | |
| NXPLWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPLWQ | 110_LOAD_RG3_TABLE | 9 | | | | 3 | | | | 2 | | | | 2304 | | | |
| NXPLWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPLWQ | 200_PROCESS_LIFT_NOTICE_RECS | 48 | | | | 10 | | | | 2 | | | | 14945472 | | | |
| NXPLWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPLWQ | 210_GETT_PROCESS_RECDS_ON_MFST | 46 | | | | 12 | | | | 2 | | | | 10731294 | | | |
| NXPLWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPLWQ | 300_PROCESS_NON_ADAM_III | 45 | | | | 10 | | | | 2 | | | | 9688320 | | | |
| NXPLWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPLWQ | 310_GETT_PROCESS_RECDS_ON_MFST | 45 | | | | 12 | | | | 2 | | | | 11250000 | | | |
| NXPLWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPLWQ | 400_CREATE_NEW_RG3_TABLE | 5 | | | | 2 | | | | 1 | | | | 320 | | | |
| NXPLWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPLWQ | 500_WRAPUP | 41 | | | | 7 | | | | 2 | | | | 2057216 | | | |
| NXPLWQ | 500_CONTINUE | 26 | | | | 2 | | | | 2 | | | | 2023866 | | | |
| NXPLWQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPLWQ | 700_UPDATE_RG3_TABLE | 13 | | | | 3 | | | | 3 | | | | 22932 | | | |
| NXPLWQ | 700_UPDATE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPLWQ | 700_ADAM_III_CHECK | 6 | | | | 4 | | | | 1 | | | | 864 | | | |
| NXPLWQ | 700_ADAM_III_CHECK_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPLWQ | 700_DETERMINE_SSCO_WRITE_REC | 57 | | | | 12 | | | | 3 | | | | 2659392 | | | |
| NXPLWQ | 700_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPLWQ | 700_ERROR_DUMP | 5 | | | | 1 | | | | 1 | | | | 405 | | | |
| NXPLWQ | 700_ERROR_DUMP_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPLWQ | 700_EOO | 11 | | | | 2 | | | | 2 | | | | 57024 | | | |
| NXPLWQ | 700_EOO_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPMWQ | 000_DRIVER | 5 | 5 | | | 4 | 4 | | | 2 | 2 | | | 20 | 20 | | |
| NXPMWQ | 000_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPMWQ | 100_R_SPAWN_RG | 26 | 26 | | | 1 | 1 | | | 1 | 1 | | | 239616 | 239616 | | |
| NXPMWQ | 100_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPMWQ | 200_PROGRAM_SELECT | 11 | 11 | | | 4 | 4 | | | 2 | 2 | | | 1584 | 1584 | | |
| NXPMWQ | 200_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPMWQ | 210_R_SPAWN_RQ | 16 | 16 | | | 2 | 2 | | | 2 | 2 | | | 97344 | 97344 | | |
| NXPMWQ | 210_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPMWQ | 211_ALLOCATE_YESTERDAY_FILE | 5 | 11 | | | 5 | 8 | | | 2 | 2 | | | 3645 | 39600 | | |
| NXPMWQ | 211_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPMWQ | 212_GET_REPLACE_CONTROL_RECORD | 46 | 46 | | | 5 | 5 | | | 2 | 2 | | | 12105774 | 12105774 | | |
| NXPMWQ | 212_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPMWQ | 213_RELEASE_YESTERDAY_FILE | 3 | 3 | | | 1 | 1 | | | 1 | 1 | | | 768 | 768 | | |
| NXPMWQ | 213_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPMWQ | 220_R_SPAWN_PL | 9 | 9 | | | 1 | 1 | | | 1 | 1 | | | 18225 | 18225 | | |
| NXPMWQ | 220_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPMWQ | 230_R_SPAWN_R2 | 9 | 9 | | | 1 | 1 | | | 1 | 1 | | | 18225 | 18225 | | |
| NXPMWQ | 230_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPMWQ | 240_R_SPAWN_R3 | 9 | 9 | | | 1 | 1 | | | 1 | 1 | | | 18225 | 18225 | | |
| NXPMWQ | 240_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPMWQ | 700_EOO | 7 | 7 | | | 2 | 2 | | | 2 | 2 | | | 7168 | 7168 | | |
| NXPMWQ | 700_EOO_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPMWQ | 700_CHECK_R_SPAWN_ERR | 5 | 5 | | | 4 | 4 | | | 2 | 2 | | | 720 | 720 | | |
| NXPMWQ | 700_CHECK_R_ERR_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPMWQ | 710_RWAIT | 3 | 3 | | | 1 | 1 | | | 1 | 1 | | | 432 | 432 | | |
| NXPMWQ | 710_RWAIT_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPMWQ | 700_ADDFIL_ERROR_CHECK | 11 | 11 | | | 3 | 3 | | | 2 | 2 | | | 33275 | 33275 | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXPMWQ | 700_ADDFIL_ERROR_CHECK_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPNWQ | 000_EOD_CONTROL | 37 | | | | 15 | | | | 2 | | | | 8739252 | | | |
| NXPNWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 100_EOD_HSKP | 67 | | | | 12 | | | | 3 | | | | 49322988 | | | |
| NXPNWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 110_LOAD_PGM_T1_TABLE | 17 | | | | 8 | | | | 3 | | | | 100793 | | | |
| NXPNWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 120_LOAD_VALID_PORT_TABLE | 5 | | | | 2 | | | | 1 | | | | 405 | | | |
| NXPNWQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 130_READ_SEQ_PA_FILE | 6 | | | | 3 | | | | 2 | | | | 54 | | | |
| NXPNWQ | 130_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 200_GET_AND_WRITE_DELETES | 36 | | | | 15 | | | | 3 | | | | 627264 | | | |
| NXPNWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 210_DELETE_OR_ABORT_PROC | 42 | | | | 9 | | | | 3 | | | | 25552800 | | | |
| NXPNWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 220_PROCESS_DELETED_TCMD | 58 | | | | 8 | | | | 2 | | | | 177219232 | | | |
| NXPNWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 250_SORT | 10 | | | | 4 | | | | 2 | | | | 6250 | | | |
| NXPNWQ | 250_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 300_GETT_AND_WRITE_LFS | 78 | | | | 23 | | | | 2 | | | | 337459200 | | | |
| NXPNWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 310_GETT_L_REC | 18 | | | | 8 | | | | 2 | | | | 56448 | | | |
| NXPNWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 311_TAB_PROCESS | 50 | | | | 7 | | | | 3 | | | | 98560800 | | | |
| NXPNWQ | 311_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 312_TCMD_PROCESS | 90 | | | | 21 | | | | 3 | | | | 876096000 | | | |
| NXPNWQ | 312_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 400_PROCESS_K1_HOLD_FILE | 18 | | | | 11 | | | | 3 | | | | 176418 | | | |
| NXPNWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 500_PROCESS_PN9_SEQUENCE_FILE | 31 | | | | 9 | | | | 3 | | | | 135036 | | | |
| NXPNWQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 800_WRAPUP_DRIVER | 6 | | | | 2 | | | | 2 | | | | 384 | | | |
| NXPNWQ | 800_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 810_WRAP_UP | 12 | | | | 7 | | | | 2 | | | | 1728 | | | |
| NXPNWQ | 810_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 820_CLOSE | 28 | | | | 5 | | | | 2 | | | | 681408 | | | |
| NXPNWQ | 820_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 830_EOO_NEXT | 21 | | | | 3 | | | | 2 | | | | 472500 | | | |
| NXPNWQ | 830_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 700_RSPAWN_PZ | 29 | | | | 7 | | | | 3 | | | | 424589 | | | |
| NXPNWQ | 700_RSPAWN_PZ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 700_WRITE_AUTODIN_HEADERS | 13 | | | | 4 | | | | 2 | | | | 39325 | | | |
| NXPNWQ | 700_WRITE_AUTODIN_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 700_COMPUTE_HOURS | 11 | | | | 5 | | | | 2 | | | | 45056 | | | |
| NXPNWQ | 700_COMPUTE_HOURS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 700_SEARCH_T1_TABLE | 7 | | | | 4 | | | | 2 | | | | 4032 | | | |
| NXPNWQ | 700_SEARCH_T1_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 700_CHECK_VALID_PORT_TBL | 5 | | | | 4 | | | | 2 | | | | 1280 | | | |
| NXPNWQ | 700_CHECK_VALID_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 700_READ_HOST | 4 | | | | 3 | | | | 2 | | | | 100 | | | |
| NXPNWQ | 700_READ_HOST_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 700_READ_SORTED_LFS | 4 | | | | 3 | | | | 2 | | | | 400 | | | |
| NXPNWQ | 700_READ_SORTED_LFS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 700_READ_K1_HOLD_FILE | 9 | | | | 4 | | | | 2 | | | | 5184 | | | |
| NXPNWQ | 700_READ_K1_HOLD_FILE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 700_HIGH_WEIGHT_ROUTINE | 15 | | | | 6 | | | | 3 | | | | 91260 | | | |
| NXPNWQ | 700_HIGH_WEIGHT_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 700_SPECIAL_PRI | 33 | | | | 32 | | | | 2 | | | | 1617 | | | |
| NXPNWQ | 700_SPECIAL_PRI_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 700_ERROR_DISPLAY | 14 | | | | 3 | | | | 2 | | | | 22400 | | | |
| NXPNWQ | 700_ERROR_DISPLAY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPNWQ | 700_SW28_RTN | 7 | | | | 3 | | | | 2 | | | | 1372 | | | |
| NXPNWQ | 700_SW28_RTN_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPPWQ | 000_DRIVER | 15 | | | | 8 | | | | 2 | | | | 10935 | | | |
| NXPPWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPPWQ | 100_HSKP | 51 | | | | 6 | | | | 2 | | | | 6247500 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXPPWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPPWQ | 110_LOAD_PGM_T1_TABLE | 18 | | | | 9 | | | | 3 | | | | 78408 | | | |
| NXPPWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPPWQ | 200_PROCESS_ADAM_III_INPUT | 47 | | | | 25 | | | | 3 | | | | 3325532 | | | |
| NXPPWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPPWQ | 210_PROCESS_TAB_RECORD | 57 | | | | 14 | | | | 2 | | | | 101130768 | | | |
| NXPPWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPPWQ | 220_PROCESS_TCMD_RECORD | 134 | | | | 45 | | | | 3 | | | | 2620247256 | | | |
| NXPPWQ | 220_CONTINUE | 23 | | | | 8 | | | | 3 | | | | 365148 | | | |
| NXPPWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPPWQ | 500_WRAPUP | 45 | | | | 5 | | | | 2 | | | | 6635520 | | | |
| NXPPWQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPPWQ | 600_SPAWN_L0 | 45 | | | | 6 | | | | 2 | | | | 4023045 | | | |
| NXPPWQ | 600_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPPWQ | 610_RSPAWN_COUNTER | 18 | | | | 6 | | | | 3 | | | | 45000 | | | |
| NXPPWQ | 610_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPPWQ | 700_COMPUTE_DATE_HR | 16 | | | | 6 | | | | 2 | | | | 40000 | | | |
| NXPPWQ | 700_COMPUTE_DATE_HR_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPPWQ | 700_HIGH_WEIGHT_ROUTINE | 14 | | | | 5 | | | | 2 | | | | 72576 | | | |
| NXPPWQ | 700_HIGH_WEIGHT_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPPWQ | 700_EDIT_DETAIL | 11 | | | | 15 | | | | 3 | | | | 1100 | | | |
| NXPPWQ | 700_EDIT_DETAIL_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPPWQ | 700_SEARCH_T1_TABLE | 7 | | | | 4 | | | | 2 | | | | 2268 | | | |
| NXPPWQ | 700_SEARCH_T1_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPPWQ | 700_READ_HOST | 4 | | | | 3 | | | | 2 | | | | 256 | | | |
| NXPPWQ | 700_READ_HOST_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPPWQ | 700_SPECIAL_PRI | 33 | | | | 32 | | | | 2 | | | | 1617 | | | |
| NXPPWQ | 700_SPECIAL_PRI_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPPWQ | 700_ERROR_DUMP | 14 | | | | 3 | | | | 2 | | | | 22400 | | | |
| NXPPWQ | 700_EXIT_ERROR_DUMP | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPQWQ | 000_EOD_CONTROL | 8 | | | | 5 | | | | 2 | | | | 800 | | | |
| NXPQWQ | 000_CONTINUE | 4 | | | | 2 | | | | 2 | | | | 36 | | | |
| NXPQWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPQWQ | 100_EOD_HSKP | 11 | | | | 2 | | | | 2 | | | | 34496 | | | |
| NXPQWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPQWQ | 200_PROCESS_ADAM_III | 25 | | | | 7 | | | | 2 | | | | 518400 | | | |
| NXPQWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPQWQ | 300_PROCESS_NON_ADAM_III | 25 | | | | 7 | | | | 2 | | | | 409600 | | | |
| NXPQWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPQWQ | 400_EOD_NEXT | 19 | | | | 2 | | | | 2 | | | | 260091 | | | |
| NXPQWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPQWQ | 500_CLOSE | 7 | | | | 2 | | | | 2 | | | | 700 | | | |
| NXPQWQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPQWQ | 700_ERROR_DISPLAY | 6 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPQWQ | 700_ERROR_DISPLAY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPRWQ | 000_IDC_CONTROL_DRIVER | 4 | | | | 2 | | | | 1 | | | | 36 | | | |
| NXPRWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPRWQ | 100_HSKP | 26 | | | | 5 | | | | 2 | | | | 988650 | | | |
| NXPRWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPRWQ | 110_BUILD_LEVEL_6_TABLE | 12 | | | | 4 | | | | 2 | | | | 6912 | | | |
| NXPRWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPRWQ | 200_CHK_WRITE_PROCESS | 12 | | | | 4 | | | | 2 | | | | 7500 | | | |
| NXPRWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPRWQ | 210_READ | 12 | | | | 6 | | | | 3 | | | | 9408 | | | |
| NXPRWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPRWQ | 220_SELECT_CONTROL | 33 | | | | 10 | | | | 2 | | | | 523908 | | | |
| NXPRWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPRWQ | 221_IDC_CATEGORY_CHK | 14 | | | | 16 | | | | 2 | | | | 504 | | | |
| NXPRWQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPRWQ | 222_TK_TYPE_CHECK | 62 | | | | 68 | | | | 5 | | | | 94302 | | | |
| NXPRWQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPRWQ | 2221_BASE_VS_CNSNE | 27 | | | | 7 | | | | 2 | | | | 771147 | | | |
| NXPRWQ | 2221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPRWQ | 230_FORMAT_CONTROL | 20 | | | | 8 | | | | 2 | | | | 30420 | | | |
| NXPRWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPRWQ | 231_RECEIPT_FORMAT | 13 | | | | 3 | | | | 2 | | | | 83200 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXPRWQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPRWQ | 232_INITIAL_FORMAT | 5 | | | | 1 | | | | 1 | | | | 320 | | | |
| NXPRWQ | 232_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPRWQ | 233_INTERMEDIATE_FORMAT | 5 | | | | 1 | | | | 1 | | | | 320 | | | |
| NXPRWQ | 233_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPRWQ | 234_FINAL_FORMAT | 7 | | | | 2 | | | | 2 | | | | 5488 | | | |
| NXPRWQ | 234_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPRWQ | 235_EXPORT_FORMAT | 18 | | | | 4 | | | | 2 | | | | 263538 | | | |
| NXPRWQ | 235_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPRWQ | 236_FORMAT_ALL | 26 | | | | 3 | | | | 2 | | | | 3313674 | | | |
| NXPRWQ | 236_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPRWQ | 300_WRAP_UP | 21 | | | | 3 | | | | 2 | | | | 774144 | | | |
| NXPRWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPVWQ | 000_DRIVER | 26 | | | | 12 | | | | 3 | | | | 665600 | | | |
| NXPVWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPVWQ | 100_HSKP | 43 | | | | 9 | | | | 3 | | | | 2730672 | | | |
| NXPVWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPVWQ | 200_REFORMAT_INPUT_FILE | 20 | | | | 9 | | | | 2 | | | | 141120 | | | |
| NXPVWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPVWQ | 210_PROCESS_PZ_FILE | 91 | | | | 16 | | | | 3 | | | | 685687275 | | | |
| NXPVWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPVWQ | 300_MERGE_INPUT_AND_PV | 27 | | | | 10 | | | | 3 | | | | 492075 | | | |
| NXPVWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPVWQ | 310_COPY_INPUT | 6 | | | | 2 | | | | 2 | | | | 864 | | | |
| NXPVWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPVWQ | 320_COPY_PV | 21 | | | | 7 | | | | 3 | | | | 365904 | | | |
| NXPVWQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPVWQ | 330_UPDATE_HEADER | 7 | | | | 2 | | | | 2 | | | | 1575 | | | |
| NXPVWQ | 330_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPVWQ | 400_FINALIZE_FILES | 22 | | | | 3 | | | | 2 | | | | 880000 | | | |
| NXPVWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPVWQ | 500_UPDATE_PV_CONTROL_REC | 35 | | | | 5 | | | | 2 | | | | 1372140 | | | |
| NXPVWQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPVWQ | 700_GETT_PV_RECORD | 12 | | | | 7 | | | | 2 | | | | 84672 | | | |
| NXPVWQ | 700_GETT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPVWQ | 700_READ_SORTED_INPUT | 11 | | | | 5 | | | | 2 | | | | 6336 | | | |
| NXPVWQ | 700_SORTED_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPVWQ | 700_WRITE_TEMP_FILE | 6 | | | | 4 | | | | 2 | | | | 1176 | | | |
| NXPVWQ | 700_WRITE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPVWQ | 700_FILE_ALLOCATE | 6 | | | | 6 | | | | 2 | | | | 6534 | | | |
| NXPVWQ | 700_FILE_ALLOCATE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPVWQ | 700_FILE_RELEASE | 4 | | | | 2 | | | | 2 | | | | 1600 | | | |
| NXPVWQ | 700_FILE_RELEASE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPVWQ | 700_ADDFIL_ERROR_CHECK | 12 | | | | 4 | | | | 2 | | | | 43200 | | | |
| NXPVWQ | 700_ADDFIL_ERROR_CHECK_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPVWQ | 700_ERROR_DISPLAY | 23 | | | | 8 | | | | 2 | | | | 3887 | | | |
| NXPVWQ | 700_ERROR_DISPLAY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPWWQ | 000_DRIVER | 4 | | | | 3 | | | | 1 | | | | 144 | | | |
| NXPWWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPWWQ | 100_HSKP | 36 | | | | 10 | | | | 2 | | | | 2304324 | | | |
| NXPWWQ | 100_CONTINUE | 3 | | | | 1 | | | | 1 | | | | 192 | | | |
| NXPWWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPWWQ | 110_T_READ_WRITE_LOOP | 19 | | | | 9 | | | | 2 | | | | 30400 | | | |
| NXPWWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPWWQ | 111_COUNTER | 11 | | | | 4 | | | | 2 | | | | 1584 | | | |
| NXPWWQ | 111_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPWWQ | 120_HOST_READ_WRITE_LOOP | 13 | | | | 3 | | | | 1 | | | | 8125 | | | |
| NXPWWQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPWWQ | 200_MERGE_COMPARE | 31 | | | | 36 | | | | 4 | | | | 2031616 | | | |
| NXPWWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPWWQ | 210_SEND_COMPOSITE_TO_ASIF | 5 | | | | 2 | | | | 1 | | | | 405 | | | |
| NXPWWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPWWQ | 220_SEND_COMPOS_CIV_TO_ASIF | 10 | | | | 3 | | | | 2 | | | | 24010 | | | |
| NXPWWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPWWQ | 230_SEND_SPECIAL_HANDLING_REC | 16 | | | | 2 | | | | 1 | | | | 389376 | | | |
| NXPWWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXPWWQ | 231_SEARCH_MONTH_TABLE | 8 | | | | 5 | | | | 2 | | | | 4608 | | | |
| NXPWWQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPWWQ | 300_WRAPUP | 32 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPWWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPWWQ | 700_READ_T_FILE | 14 | | | | 7 | | | | 2 | | | | 115934 | | | |
| NXPWWQ | 700_READ_T_FILE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPWWQ | 700_READ_HOST_FILE | 12 | | | | 3 | | | | 2 | | | | 76800 | | | |
| NXPWWQ | 700_READ_HOST_FILE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPYWQ | 000_DRIVER | 19 | 19 | | | 10 | 10 | | | 2 | 2 | | | 260091 | 260091 | | |
| NXPYWQ | 000_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPYWQ | 100_HSKP | 43 | 43 | | | 9 | 9 | | | 3 | 3 | | | 2730672 | 2730672 | | |
| NXPYWQ | 100_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPYWQ | 200_REFORMAT_INPUT_FILE | 17 | 20 | | | 7 | 9 | | | 2 | 2 | | | 88128 | 162000 | | |
| NXPYWQ | 200_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPYWQ | 210_PROCESS_PZ_FILE | 91 | 91 | | | 16 | 16 | | | 3 | 3 | | | 685687275 | 685687275 | | |
| NXPYWQ | 210_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPYWQ | 300_MERGE_INPUT_AND_PY | 27 | 27 | | | 10 | 10 | | | 3 | 3 | | | 492075 | 492075 | | |
| NXPYWQ | 300_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPYWQ | 310_COPY_INPUT | 6 | 6 | | | 2 | 2 | | | 2 | 2 | | | 864 | 864 | | |
| NXPYWQ | 310_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPYWQ | 320_COPY_PY | 21 | 21 | | | 7 | 7 | | | 3 | 3 | | | 365904 | 365904 | | |
| NXPYWQ | 320_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPYWQ | 330_UPDATE_HEADER | 7 | 7 | | | 2 | 2 | | | 2 | 2 | | | 1575 | 1575 | | |
| NXPYWQ | 330_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPYWQ | 400_FINALIZE_FILES | 22 | 22 | | | 3 | 3 | | | 2 | 2 | | | 880000 | 880000 | | |
| NXPYWQ | 400_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPYWQ | 700_GETT_PY_RECORD | 12 | 12 | | | 7 | 7 | | | 2 | 2 | | | 84672 | 84672 | | |
| NXPYWQ | 700_GETT_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPYWQ | 700_READ_SORTED_INPUT | 10 | 10 | | | 5 | 5 | | | 2 | 2 | | | 7290 | 7290 | | |
| NXPYWQ | 700_SORTED_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPYWQ | 700_WRITE_TEMP_FILE | 6 | 6 | | | 4 | 4 | | | 2 | 2 | | | 1176 | 1176 | | |
| NXPYWQ | 700_WRITE_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPYWQ | 700_FILE_ALLOCATE | 6 | 6 | | | 6 | 6 | | | 2 | 2 | | | 6534 | 6534 | | |
| NXPYWQ | 700_FILE_ALLOCATE_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPYWQ | 700_FILE_RELEASE | 4 | 4 | | | 2 | 2 | | | 2 | 2 | | | 1600 | 1600 | | |
| NXPYWQ | 700_FILE_RELEASE_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPYWQ | 700_ADDFIL_ERROR_CHECK | 12 | 12 | | | 4 | 4 | | | 2 | 2 | | | 43200 | 43200 | | |
| NXPYWQ | 700_ADDFIL_ERROR_CHECK_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPYWQ | 700_ERROR_DISPLAY | 23 | 23 | | | 8 | 8 | | | 2 | 2 | | | 3312 | 3312 | | |
| NXPYWQ | 700_ERROR_DISPLAY_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXPZWQ | 000_DRIVER | 9 | | | | 6 | | | | 2 | | | | 21609 | | | |
| NXPZWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 100_UPDT_EOD_HSKP | 56 | | | | 16 | | | | 3 | | | | 39513600 | | | |
| NXPZWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 110_ADAM_III_INPUT | 34 | | | | 20 | | | | 5 | | | | 1470976 | | | |
| NXPZWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 120_ASIF_INPUT | 26 | | | | 8 | | | | 3 | | | | 733824 | | | |
| NXPZWQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 130_LOAD_MONTH_YEAR_TABLE | 1 | | | | 3 | | | | 1 | | | | 36 | | | |
| NXPZWQ | 130_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 131_LOAD_MONTH_TABLE | 5 | | | | 2 | | | | 1 | | | | 180 | | | |
| NXPZWQ | 131_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 200_MERGE_COMPARE | 27 | | | | 34 | | | | 2 | | | | 292032 | | | |
| NXPZWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 210_DELETED_TCMD_SORTED | 5 | | | | 1 | | | | 1 | | | | 720 | | | |
| NXPZWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 220_DELETED_TCMD_HIST | 6 | | | | 1 | | | | 1 | | | | 600 | | | |
| NXPZWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 230_COPY_SORTED | 9 | | | | 9 | | | | 2 | | | | 97344 | | | |
| NXPZWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 240_COPY_HIST | 6 | | | | 1 | | | | 1 | | | | 600 | | | |
| NXPZWQ | 240_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 250_ABORT_SORTED_RECS | 7 | | | | 1 | | | | 1 | | | | 7168 | | | |
| NXPZWQ | 250_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 260_ABORT_HIST_RECORDS | 17 | | | | 20 | | | | 3 | | | | 205700 | | | |
| NXPZWQ | 260_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXPZWQ | 261_READ_SORTED | 5 | | | | 2 | | | | 1 | | | | 180 | | | |
| NXPZWQ | 261_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 270_DELETE_REC_WRITE | 5 | | | | 1 | | | | 1 | | | | 720 | | | |
| NXPZWQ | 270_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 280_WRITE_AND_ABORT_RECORDS | 18 | | | | 6 | | | | 2 | | | | 352800 | | | |
| NXPZWQ | 280_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 281_WRITE_SORTED_TO_HIST | 14 | | | | 8 | | | | 2 | | | | 236600 | | | |
| NXPZWQ | 281_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 300_WRAP_UP | 14 | | | | 4 | | | | 2 | | | | 98784 | | | |
| NXPZWQ | 300_CONTINUE | 15 | | | | 6 | | | | 2 | | | | 1215 | | | |
| NXPZWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 310_TOTAL_DISPLAYS | 40 | | | | 2 | | | | 1 | | | | 1713960 | | | |
| NXPZWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 320_UPDATE_RANDOM_FILES | 10 | | | | 2 | | | | 1 | | | | 1440 | | | |
| NXPZWQ | 320_CONTINUE | 10 | | | | 4 | | | | 2 | | | | 4000 | | | |
| NXPZWQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 321_Z3_REWRITE | 5 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 321_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 330_GETT_L0_CONTROL_RECORD | 25 | | | | 3 | | | | 2 | | | | 342225 | | | |
| NXPZWQ | 330_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 340_RSPAWN_PARAGRAPH | 35 | | | | 6 | | | | 2 | | | | 56000 | | | |
| NXPZWQ | 340_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 341_RSPAWN_CALL | 8 | | | | 2 | | | | 2 | | | | 12800 | | | |
| NXPZWQ | 341_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 350_UPDATE_L0_CONTROL_RECORD | 19 | | | | 3 | | | | 2 | | | | 98496 | | | |
| NXPZWQ | 350_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 360_MOVE_TABLE_TO_PZ2 | 1 | | | | 2 | | | | 1 | | | | 16 | | | |
| NXPZWQ | 360_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 361_MOVE_MONTHLY_DATA | 2 | | | | 2 | | | | 1 | | | | 128 | | | |
| NXPZWQ | 361_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 700_TOP_OF_PAGE | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 700_TOP_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 700_DISPLAY_COUNTERS | 24 | | | | 4 | | | | 2 | | | | 6144 | | | |
| NXPZWQ | 700_DISPLAY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 700_READ_OLD_HIST | 6 | | | | 2 | | | | 1 | | | | 3456 | | | |
| NXPZWQ | 700_READ_OLD_HIST_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 700_READ_SORT_FILE | 22 | | | | 9 | | | | 2 | | | | 266200 | | | |
| NXPZWQ | 700_READ_SORT_FILE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 700_FIND_LIFT_MONTH | 11 | | | | 7 | | | | 2 | | | | 27500 | | | |
| NXPZWQ | 700_FIND_LIFT_MONTH_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 700_WRITE_HISTORY | 5 | | | | 5 | | | | 2 | | | | 6480 | | | |
| NXPZWQ | 700_WRITE_HISTORY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 700_OLD_MONTH_PROC | 35 | | | | 15 | | | | 3 | | | | 4046000 | | | |
| NXPZWQ | 700_OLD_MONTH_PROC_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 700_Z3_REWRITE | 5 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 700_Z3_REWRITE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 700_WRITE_Z1 | 6 | | | | 2 | | | | 1 | | | | 6144 | | | |
| NXPZWQ | 700_WRITE_Z1_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 700_ERROR_DISPLAY | 8 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXPZWQ | 700_ERROR_DISPLAY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR0WQ | 000_DRIVER | 16 | | | | 7 | | | | 2 | | | | 176400 | | | |
| NXR0WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR0WQ | 100_HOUSEKEEPING | 44 | | | | 8 | | | | 2 | | | | 3933644 | | | |
| NXR0WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR0WQ | 110_READ_WRITE | 5 | | | | 3 | | | | 1 | | | | 180 | | | |
| NXR0WQ | 110_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| NXR0WQ | 200_CLEAR_OLD | 12 | | | | 9 | | | | 3 | | | | 10800 | | | |
| NXR0WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR0WQ | 300_PROCESS_INPUT | 4 | | | | 6 | | | | 2 | | | | 400 | | | |
| NXR0WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR0WQ | 310_PROCESS_NEW | 19 | | | | 6 | | | | 2 | | | | 45619 | | | |
| NXR0WQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR0WQ | 320_PROCESS_ERROR | 20 | | | | 6 | | | | 2 | | | | 81920 | | | |
| NXR0WQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR0WQ | 400_SORT_MERGE_T1_T2 | 3 | | | | 2 | | | | 2 | | | | 108 | | | |
| NXR0WQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXR0WQ | 410_READ_INPUT_FILES_PARA | 6 | | | | 4 | | | | 2 | | | | 864 | | | |
| NXR0WQ | 411_READ_T1_TEMP | 6 | | | | 3 | | | | 2 | | | | 54 | | | |
| NXR0WQ | 411_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR0WQ | 412_READ_TEMP_INPUT | 7 | | | | 4 | | | | 2 | | | | 112 | | | |
| NXR0WQ | 412_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR0WQ | 410_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR0WQ | 420_WRITE_R0_PARA | 12 | | | | 5 | | | | 2 | | | | 24300 | | | |
| NXR0WQ | 421_RETURN_WRITE_REC | 14 | | | | 5 | | | | 3 | | | | 3150 | | | |
| NXR0WQ | 421_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR0WQ | 420_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR0WQ | 500_EOO_WRAP_UP | 12 | | | | 3 | | | | 2 | | | | 52272 | | | |
| NXR0WQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR0WQ | 700_EDIT_INPUT | 89 | | | | 35 | | | | 3 | | | | 118933904 | | | |
| NXR0WQ | 700_EDIT_INPUT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR0WQ | 700_FORMAT_AND_WRITE | 119 | | | | 106 | | | | 2 | | | | 8044400 | | | |
| NXR0WQ | 700_FORMAT_AND_WRITE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 000_DRIVER | 16 | | | | 6 | | | | 2 | | | | 132496 | | | |
| NXR1WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 100_HOUSEKEEPING | 30 | | | | 6 | | | | 2 | | | | 6021120 | | | |
| NXR1WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 200_SELECT_RECORDS_FOR_RPT | 5 | | | | 5 | | | | 2 | | | | 720 | | | |
| NXR1WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 210_READ_R0 | 8 | | | | 4 | | | | 2 | | | | 512 | | | |
| NXR1WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 220_PROCESS_SELECTED_RECORD | 11 | | | | 6 | | | | 2 | | | | 19404 | | | |
| NXR1WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 300_WRITE_REPORT | 17 | | | | 5 | | | | 2 | | | | 103428 | | | |
| NXR1WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 310_READ_FIRST_TIME | 10 | | | | 4 | | | | 2 | | | | 20250 | | | |
| NXR1WQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 320_PROCESS_RECORD | 35 | | | | 7 | | | | 2 | | | | 5826240 | | | |
| NXR1WQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 330_PROCESS_SUB_AREA_TOTALS | 34 | | | | 9 | | | | 2 | | | | 10359936 | | | |
| NXR1WQ | 330_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 340_PROCESS_MAI_AREA_TOTALS | 43 | | | | 9 | | | | 2 | | | | 21070000 | | | |
| NXR1WQ | 340_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 350_PROCESS_TYPE_TOTALS | 53 | | | | 13 | | | | 2 | | | | 53851392 | | | |
| NXR1WQ | 350_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 360_PROCESS_MFST_STA_TOTALS | 45 | | | | 8 | | | | 2 | | | | 29963520 | | | |
| NXR1WQ | 360_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 370_PROCESS_NAF_TOTALS | 124 | | | | 25 | | | | 2 | | | | 882659776 | | | |
| NXR1WQ | 370_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 380_PROCESS_MAC_TOTALS | 118 | | | | 23 | | | | 2 | | | | 776348550 | | | |
| NXR1WQ | 380_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 700_MAI_TABLE_SEARCH | 9 | | | | 4 | | | | 2 | | | | 4356 | | | |
| NXR1WQ | 700_MAI_SEARCH_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 700_TYPE_SEARCH | 6 | | | | 4 | | | | 2 | | | | 4056 | | | |
| NXR1WQ | 700_TYPE_SEARCH_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 700_WRITE_HEADERS | 18 | | | | 9 | | | | 2 | | | | 526338 | | | |
| NXR1WQ | 700_WRITE_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 700_WRITE_TRAILER | 7 | | | | 3 | | | | 2 | | | | 25200 | | | |
| NXR1WQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR1WQ | 700_ERROR_PROCESS | 6 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXR1WQ | 700_ERROR_PROCESS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR2WQ | 000_CONTROL | 14 | | | | 7 | | | | 2 | | | | 6174 | | | |
| NXR2WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR2WQ | 100_HSKP | 50 | | | | 9 | | | | 2 | | | | 6372450 | | | |
| NXR2WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR2WQ | 110_BUILD_ABORTED_MFST_TABLE | 15 | | | | 7 | | | | 2 | | | | 138240 | | | |
| NXR2WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR2WQ | 120_LOAD_PGM_T1_TABLE | 21 | | | | 9 | | | | 3 | | | | 124509 | | | |
| NXR2WQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR2WQ | 200_GETT_AND_RITE_ADAM_III_LFS | 72 | | | | 23 | | | | 2 | | | | 233280000 | | | |
| NXR2WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR2WQ | 210_GETT_L_REC | 30 | | | | 10 | | | | 2 | | | | 1642680 | | | |
| NXR2WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXR2WQ | 211_TAB_PROCESS | 12 | | | | 2 | | | | 2 | | | | 117612 | | | |
| NXR2WQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR2WQ | 212_TCMD_PROCESS | 44 | | | | 10 | | | | 2 | | | | 21314304 | | | |
| NXR2WQ | 212_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR2WQ | 300_WRAP_UP | 32 | | | | 7 | | | | 2 | | | | 1620000 | | | |
| NXR2WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR2WQ | 700_SEARCH_T1_TABLE | 7 | | | | 6 | | | | 2 | | | | 1372 | | | |
| NXR2WQ | 700_SEARCH_T1_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR2WQ | 700_SEARCH_T1_TABLE_SEQ | 7 | | | | 6 | | | | 2 | | | | 4032 | | | |
| NXR2WQ | 700_SEARCH_T1_TABLE_SEQ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR2WQ | 700_HIGH_WEIGHT_ROUTINE | 14 | | | | 5 | | | | 2 | | | | 72576 | | | |
| NXR2WQ | 700_HIGH_WEIGHT_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR2WQ | 700_ERROR_DISPLAY | 9 | | | | 2 | | | | 2 | | | | 1764 | | | |
| NXR2WQ | 700_ERROR_DISPLAY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR3WQ | 000_CONTROL | 12 | | | | 7 | | | | 2 | | | | 5292 | | | |
| NXR3WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR3WQ | 100_HSKP | 46 | | | | 7 | | | | 2 | | | | 7323246 | | | |
| NXR3WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR3WQ | 110_LOAD_TABLES | 19 | | | | 10 | | | | 3 | | | | 121600 | | | |
| NXR3WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR3WQ | 200_PROCESS_ADAM_III_INPUT | 48 | | | | 18 | | | | 3 | | | | 10830000 | | | |
| NXR3WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR3WQ | 210_PROCESS_TAB_RECORD | 6 | | | | 8 | | | | 2 | | | | 216 | | | |
| NXR3WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR3WQ | 220_PROCESS_TCMD_RECORD | 54 | | | | 25 | | | | 2 | | | | 30699864 | | | |
| NXR3WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR3WQ | 300_WRAP_UP | 31 | | | | 4 | | | | 2 | | | | 2430400 | | | |
| NXR3WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR3WQ | 700_COMPUTE_HOURS | 9 | | | | 5 | | | | 2 | | | | 6561 | | | |
| NXR3WQ | 700_COMPUTE_HOURS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR3WQ | 700_HIGH_WEIGHT_ROUTINE | 14 | | | | 5 | | | | 2 | | | | 50400 | | | |
| NXR3WQ | 700_HIGH_WEIGHT_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR3WQ | 700_WRITE_T1 | 28 | | | | 7 | | | | 2 | | | | 1612800 | | | |
| NXR3WQ | 700_WRITE_T1_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR3WQ | 700_SEARCH_STN_TABLE | 7 | | | | 4 | | | | 2 | | | | 1372 | | | |
| NXR3WQ | 700_SEARCH_STN_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR3WQ | 700_SEARCH_STN_TABLE_SEQ | 7 | | | | 6 | | | | 2 | | | | 4032 | | | |
| NXR3WQ | 700_SEARCH_STN_TABLE_SEQ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR3WQ | 700_SEARCH_CHNL_TABLE | 6 | | | | 4 | | | | 2 | | | | 864 | | | |
| NXR3WQ | 700_SEARCH_CHNL_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR3WQ | 700_CHECK_NAF_INDICATORS | 21 | | | | 6 | | | | 2 | | | | 25725 | | | |
| NXR3WQ | 700_CHECK_NAF_INDICATORS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR3WQ | 700_ERROR_DISPLAY | 7 | | | | 2 | | | | 2 | | | | 1008 | | | |
| NXR3WQ | 700_ERR_DISPLAY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR5WQ | 000_DRIVER | 5 | | | | 2 | | | | 1 | | | | 80 | | | |
| NXR5WQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR5WQ | 100_HSKP | 8 | | | | 2 | | | | 1 | | | | 33800 | | | |
| NXR5WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR5WQ | 110_READ_WRITE_TO_TEMP | 9 | | | | 3 | | | | 1 | | | | 3600 | | | |
| NXR5WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR5WQ | 200_PROCESS | 32 | | | | 15 | | | | 3 | | | | 627200 | | | |
| NXR5WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR5WQ | 300_RESORT_RECS | 5 | | | | 2 | | | | 1 | | | | 980 | | | |
| NXR5WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR5WQ | 310_READ_WRITE | 7 | | | | 3 | | | | 1 | | | | 1008 | | | |
| NXR5WQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR5WQ | 400_WRAPUP | 14 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXR5WQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR6WQ | 000_DRIVER | 4 | | | | 2 | | | | 1 | | | | 36 | | | |
| NXR6WQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR6WQ | 100_HSKP | 3 | | | | 1 | | | | 1 | | | | 192 | | | |
| NXR6WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR6WQ | 200_READ_WRITE | 11 | | | | 5 | | | | 2 | | | | 17600 | | | |
| NXR6WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR6WQ | 300_WRAPUP | 3 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXR6WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | January | April | Baseline | October | January | April | Baseline | October | January | April | Baseline | October | January | April |
| NXR7WQ | 000_HSKP | 8 | 8 | | | 2 | 2 | | | 2 | 2 | | | 4608 | 4608 | | |
| NXR7WQ | 000_CLOSE | 8 | 9 | | | 2 | 2 | | | 2 | 2 | | | 5832 | 8100 | | |
| NXR7WQ | 400_STARS_FILE_CK | 5 | 5 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXR7WQ | 450_STARS_FILE_CK | 6 | 6 | | | 3 | 3 | | | 2 | 2 | | | 216 | 216 | | |
| NXR7WQ | 500_TCN_CHECK | 9 | 10 | | | 4 | 4 | | | 2 | 2 | | | 2304 | 4000 | | |
| NXRAWQ | 000_CONTROL | 5 | | | | 3 | | | | 2 | | | | 20 | | | |
| NXRAWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRAWQ | 100_TC_MOVEMENT | 15 | | | | 5 | | | | 2 | | | | 30375 | | | |
| NXRAWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRAWQ | 110_READ_WRITE_LOOP | 9 | | | | 3 | | | | 1 | | | | 729 | | | |
| NXRAWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRAWQ | 200_TC_ONHAND | 12 | | | | 4 | | | | 1 | | | | 9408 | | | |
| NXRAWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRAWQ | 210_READ_WRITE_LOOP | 9 | | | | 3 | | | | 1 | | | | 729 | | | |
| NXRAWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRDWQ | 000_CONTROL | 13 | | | | 5 | | | | 2 | | | | 5200 | | | |
| NXRDWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRDWQ | 100_CHECK_NUM_RECORDS | 6 | | | | 2 | | | | 1 | | | | 216 | | | |
| NXRDWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRDWQ | 200_BUILD | 6 | | | | 3 | | | | 2 | | | | 216 | | | |
| NXRDWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRDWQ | 210_BUILD_MV | 9 | | | | 2 | | | | 1 | | | | 324 | | | |
| NXRDWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRDWQ | 220_BUILD_OH | 9 | | | | 2 | | | | 1 | | | | 324 | | | |
| NXRDWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRDWQ | 300_DUMP_ONE_RECORD | 6 | | | | 3 | | | | 2 | | | | 216 | | | |
| NXRDWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRDWQ | 310_DUMP_ONE_T_MV_REC | 9 | | | | 2 | | | | 1 | | | | 144 | | | |
| NXRDWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRDWQ | 320_DUMP_ONE_T_OH_REC | 9 | | | | 2 | | | | 1 | | | | 144 | | | |
| NXRDWQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRDWQ | 400_T_COMPOSITE_MV | 14 | | | | 9 | | | | 2 | | | | 18144 | | | |
| NXRDWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRDWQ | 410_T_MV_READ_LOOP | 7 | | | | 2 | | | | 1 | | | | 448 | | | |
| NXRDWQ | 410_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRDWQ | 500_T_COMPOSITE_OH | 14 | | | | 9 | | | | 2 | | | | 18144 | | | |
| NXRDWQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRDWQ | 510_T_OH_READ_LOOP | 7 | | | | 2 | | | | 1 | | | | 448 | | | |
| NXRDWQ | 510_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXREWQ | 000_DRIVER | 4 | | | | 2 | | | | 1 | | | | 36 | | | |
| NXREWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXREWQ | 100_HSKP | 10 | | | | 2 | | | | 1 | | | | 20250 | | | |
| NXREWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXREWQ | 110_MAKE_TEMP_FILE | 6 | | | | 3 | | | | 1 | | | | 486 | | | |
| NXREWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXREWQ | 200_PROCESS | 14 | | | | 28 | | | | 3 | | | | 89600 | | | |
| NXREWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXREWQ | 300_WRAPUP | 9 | | | | 1 | | | | 1 | | | | 729 | | | |
| NXREWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXREWQ | 700_READ_INPUT | 7 | | | | 3 | | | | 2 | | | | 1008 | | | |
| NXREWQ | 700_READ_INPUT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXREWQ | 700_READ_MOVE | 9 | | | | 4 | | | | 2 | | | | 5625 | | | |
| NXREWQ | 700_READ_MOVE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRGWQ | 000_MAIN_DRIVER | 18 | 18 | | | 10 | 10 | | | 2 | 2 | | | 176418 | 176418 | | |
| NXRGWQ | 000_STOP | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 100_HSKP | 30 | 30 | | | 8 | 8 | | | 2 | 2 | | | 1297920 | 1297920 | | |
| NXRGWQ | 100_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 110_VALID_RPT | 4 | 4 | | | 3 | 3 | | | 1 | 1 | | | 576 | 576 | | |
| NXRGWQ | 110_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 111_LOAD_VALID_RPT_TABLE | 2 | 2 | | | 1 | 1 | | | 1 | 1 | | | 72 | 72 | | |
| NXRGWQ | 111_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 115_VALID_HQS_RPT | 7 | 7 | | | 4 | 4 | | | 2 | 2 | | | 700 | 700 | | |
| NXRGWQ | 115_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 1151_RG4_REPORT_TABLE | 7 | 7 | | | 3 | 3 | | | 1 | 1 | | | 2268 | 2268 | | |
| NXRGWQ | 1151_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 1152_RG6_EOM_RPT_TABLE | 7 | 7 | | | 3 | 3 | | | 1 | 1 | | | 2268 | 2268 | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXRGWQ | 1152_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 116_LOAD_VALID_HQS_RPT_TABLE | 3 | 3 | | | 1 | 1 | | | 1 | 1 | | | 108 | 108 | | |
| NXRGWQ | 116_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 117_LOAD_VALID_HQS_TABLE_RG6 | 3 | 3 | | | 1 | 1 | | | 1 | 1 | | | 108 | 108 | | |
| NXRGWQ | 117_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 120_INPUT_READ | 8 | 8 | | | 2 | 2 | | | 1 | 1 | | | 648 | 648 | | |
| NXRGWQ | 120_INPUT_READ_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 200_PROCESS | 21 | 21 | | | 9 | 9 | | | 2 | 2 | | | 537600 | 537600 | | |
| NXRGWQ | 200_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 210_INPUT_PROCESS | 24 | 24 | | | 8 | 8 | | | 3 | 3 | | | 221184 | 221184 | | |
| NXRGWQ | 210_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 211_CHECK_REPORTS | 10 | 10 | | | 6 | 6 | | | 2 | 2 | | | 24010 | 24010 | | |
| NXRGWQ | 211_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 212_CHECK_HQS | 20 | 20 | | | 9 | 9 | | | 2 | 2 | | | 242000 | 242000 | | |
| NXRGWQ | 212_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 213_CHECK_ASIF | 20 | 20 | | | 9 | 9 | | | 2 | 2 | | | 242000 | 242000 | | |
| NXRGWQ | 213_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 220_PROCESS | 27 | 32 | | | 12 | 15 | | | 2 | 2 | | | 1179387 | 2230272 | | |
| NXRGWQ | 220_CONTINUE | 15 | 15 | | | 3 | 3 | | | 2 | 2 | | | 188160 | 188160 | | |
| NXRGWQ | 220_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 221_OUTPUT_PROCESS | 8 | 8 | | | 5 | 5 | | | 2 | 2 | | | 4608 | 4608 | | |
| NXRGWQ | 221_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 221_SKIP | 4 | 4 | | | 4 | 4 | | | 1 | 1 | | | 400 | 400 | | |
| NXRGWQ | 221_SKIP_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 221_SELECT | 4 | 4 | | | 4 | 4 | | | 1 | 1 | | | 400 | 400 | | |
| NXRGWQ | 221_SELECT_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 2211_SKIP_RECORD | 2 | 2 | | | 2 | 2 | | | 1 | 1 | | | 8 | 8 | | |
| NXRGWQ | 2211_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 2212_WRITE_OUTPUT_REC | 5 | 5 | | | 3 | 3 | | | 1 | 1 | | | 720 | 720 | | |
| NXRGWQ | 2212_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 700_CHECK_STATION_CODE | 16 | 16 | | | 5 | 5 | | | 3 | 3 | | | 123904 | 123904 | | |
| NXRGWQ | 700_CHECK_STATION_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 700_READ_TERMINAL_IDS | 9 | 9 | | | 4 | 4 | | | 2 | 2 | | | 7056 | 7056 | | |
| NXRGWQ | 700_READ_TERMINAL_IDS_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 700_SET_UP_ALL_STATIONS | 7 | 7 | | | 4 | 4 | | | 2 | 2 | | | 4032 | 4032 | | |
| NXRGWQ | 700_EXIT_SET_UP_ALL_STATIONS | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 700_LOAD_STATION_HOLD_TABLE | 3 | 3 | | | 1 | 1 | | | 1 | 1 | | | 108 | 108 | | |
| NXRGWQ | 700_EXIT_STATION_HOLD_TABLE | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 700_ADDFIL_ERROR_CHECK | 12 | 12 | | | 3 | 3 | | | 2 | 2 | | | 62208 | 62208 | | |
| NXRGWQ | 700_ADDFIL_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 700_SEARCH_PORT_RPT_TABLE | 13 | 13 | | | 4 | 4 | | | 2 | 2 | | | 22932 | 22932 | | |
| NXRGWQ | 700_SEARCH_PORT_RPT_TABLE_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 700_SEARCH_HQS_RPT_TABLE | 13 | 13 | | | 4 | 4 | | | 2 | 2 | | | 16848 | 16848 | | |
| NXRGWQ | 700_SEARCH_HQS_RPT_TABLE_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 700_INPUT_ERROR | 11 | 11 | | | 1 | 1 | | | 1 | 1 | | | 174636 | 174636 | | |
| NXRGWQ | 700_INPUT_ERROR_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 700_ERROR_DUMP | 3 | 3 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 700_ERROR_DUMP_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 700_INPUT_READ | 7 | 7 | | | 2 | 2 | | | 1 | 1 | | | 252 | 252 | | |
| NXRGWQ | 700_INPUT_READ_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRGWQ | 700_SEND_MESSAGE | 40 | 40 | | | 7 | 7 | | | 2 | 2 | | | 5416960 | 5416960 | | |
| NXRGWQ | 700_SEND_MESSAGE_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXRIWQ | 000_DRIVER | 4 | | | | 2 | | | | 2 | | | | 16 | | | |
| NXRIWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRIWQ | 100_TC_ONHAND | 12 | | | | 3 | | | | 1 | | | | 9408 | | | |
| NXRIWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRIWQ | 110_READ_WRITE_LOOP | 12 | | | | 4 | | | | 2 | | | | 21168 | | | |
| NXRIWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRIWQ | 200_TC_MOVEMENT | 12 | | | | 3 | | | | 1 | | | | 9408 | | | |
| NXRIWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRIWQ | 210_READ_WRITE_LOOP | 12 | | | | 4 | | | | 2 | | | | 21168 | | | |
| NXRIWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRKWQ | 000_DRIVER | 4 | | | | 2 | | | | 2 | | | | 16 | | | |
| NXRKWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRKWQ | 100_TC_ONHAND | 11 | | | | 3 | | | | 1 | | | | 8624 | | | |
| NXRKWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXRKWQ | 110_READ_WRITE_LOOP | 13 | | | | 4 | | | | 2 | | | | 11700 | | | |
| NXRKWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRKWQ | 200_TC_MOVEMENT | 11 | | | | 3 | | | | 1 | | | | 8624 | | | |
| NXRKWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRKWQ | 210_READ_WRITE_LOOP | 13 | | | | 4 | | | | 2 | | | | 11700 | | | |
| NXRKWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRMWQ | 000_DRIVER | 4 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRMWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRMWQ | 100_HSKP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRMWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRMWQ | 200_ADD_OR_DELETE | 10 | | | | 9 | | | | 2 | | | | 3240 | | | |
| NXRMWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRMWQ | 210_DELETE_MAI | 14 | | | | 5 | | | | 2 | | | | 5600 | | | |
| NXRMWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRMWQ | 220_ADD_MAI | 23 | | | | 8 | | | | 2 | | | | 46575 | | | |
| NXRMWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRMWQ | 300_WRAPUP | 7 | | | | 3 | | | | 2 | | | | 252 | | | |
| NXRMWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRMWQ | 310_TEMP_TO_MAI | 6 | | | | 3 | | | | 1 | | | | 384 | | | |
| NXRMWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRMWQ | 700_READ_WRITE_MAI | 5 | | | | 3 | | | | 1 | | | | 320 | | | |
| NXRMWQ | 700_READ_WRITE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRMWQ | 700_READ_MAI | 3 | | | | 2 | | | | 1 | | | | 27 | | | |
| NXRMWQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRPWQ | 000_DRIVER | 8 | 8 | | 11 | 3 | 3 | | 5 | 2 | 2 | | 2 | 4608 | 4608 | | 13475 |
| NXRPWQ | 000_STOP | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| NXRPWQ | 100_HSKP | 25 | 25 | | 27 | 7 | 7 | | 9 | 2 | 2 | | 2 | 608400 | 608400 | | 762048 |
| NXRPWQ | 100_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| NXRPWQ | 110_VALIDATE_REPORT_CODE | 14 | 14 | | 16 | 6 | 6 | | 8 | 2 | 2 | | 2 | 14336 | 14336 | | 20736 |
| NXRPWQ | 110_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| NXRPWQ | 120_LOAD_REPORT_STATION_TABLE | 12 | 12 | | 12 | 3 | 3 | | 3 | 2 | 2 | | 2 | 10800 | 10800 | | 10800 |
| NXRPWQ | 120_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| NXRPWQ | 130_INITIALIZE_REPORT_FILE | 18 | 18 | | 19 | 6 | 6 | | 7 | 2 | 2 | | 2 | 16200 | 16200 | | 23275 |
| NXRPWQ | 130_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| NXRPWQ | 131_ADDFIL_CALL | 8 | 13 | | 14 | 5 | 8 | | 9 | 2 | 2 | | 2 | 24200 | 83200 | | 101150 |
| NXRPWQ | 131_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| NXRPWQ | 200_PROCESS_REPORT_FILE | 7 | 7 | | 8 | 4 | 4 | | 5 | 1 | 1 | | 2 | 2800 | 2800 | | 4608 |
| NXRPWQ | 200_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| NXRPWQ | 200_SEARCH_AT_END | 3 | 3 | | 4 | 4 | 4 | | 5 | 1 | 1 | | 2 | 300 | 300 | | 576 |
| NXRPWQ | 200_SEARCH_WHEN_1 | 2 | 2 | | 3 | 2 | 2 | | 3 | 1 | 1 | | 2 | 8 | 8 | | 27 |
| NXRPWQ | 200_SEARCH_WHEN_2 | 4 | 4 | | 5 | 4 | 4 | | 5 | 1 | 1 | | 2 | 900 | 900 | | 1620 |
| NXRPWQ | 210_WRITE_AND_READ_NEXT | 7 | 7 | | 9 | 3 | 3 | | 5 | 2 | 2 | | 2 | 2268 | 2268 | | 3969 |
| NXRPWQ | 210_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| NXRPWQ | 220_SKIP_THIS_STATION | 4 | 4 | | 6 | 2 | 2 | | 4 | 1 | 1 | | 2 | 36 | 36 | | 96 |
| NXRPWQ | 220_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| NXRPWQ | 300_CLOSE | 11 | 11 | | 13 | 5 | 5 | | 7 | 2 | 2 | | 2 | 70400 | 70400 | | 93925 |
| NXRPWQ | 300_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| NXRPWQ | 310_DISPLAY | 8 | 8 | | 9 | 2 | 2 | | 3 | 2 | 2 | | 2 | 512 | 512 | | 900 |
| NXRPWQ | 310_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| NXRPWQ | 700_ADDFIL_ERROR_CHECK | 11 | 11 | | 12 | 3 | 3 | | 4 | 2 | 2 | | 2 | 33275 | 33275 | | 43200 |
| NXRPWQ | 700_ADDFIL_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| NXRPWQ | 700_READ_REQUEST_FILE | 4 | 4 | | 6 | 2 | 2 | | 4 | 1 | 1 | | 2 | 64 | 64 | | 150 |
| NXRPWQ | 700_READ_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| NXRQWQ | 000_DRIVER | 4 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRQWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRQWQ | 100_HOUSEKEEPING | 8 | | | | 1 | | | | 1 | | | | 28800 | | | |
| NXRQWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRQWQ | 200_WRITE_BANNER_PAGE | 53 | | | | 53 | | | | 1 | | | | 2597 | | | |
| NXRQWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRQWQ | 300_WRAPUP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRQWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRRWQ | 000_DRIVER | 8 | | | | 3 | | | | 2 | | | | 128 | | | |
| NXRRWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRRWQ | 100_HOUSEKEEPING | 14 | | | | 3 | | | | 1 | | | | 274400 | | | |
| NXRRWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRRWQ | 200_WRITE_BANNER_PAGE | 5 | | | | 3 | | | | 1 | | | | 180 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXRRWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRRWQ | 300_WRITE_MSG_PAGE | 20 | | | | 6 | | | | 2 | | | | 11520 | | | |
| NXRRWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRRWQ | 310_WRITE_MSG | 6 | | | | 3 | | | | 1 | | | | 1536 | | | |
| NXRRWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRRWQ | 400_WRAPUP | 7 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXRRWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRSWQ | 000_DRIVER | 20 | | | | 8 | | | | 2 | | | | 18000 | | | |
| NXRSWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRSWQ | 100_HSKP | 9 | | | | 2 | | | | 2 | | | | 15876 | | | |
| NXRSWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRSWQ | 200_GET_CONTROL_RECORD | 13 | | | | 2 | | | | 2 | | | | 63700 | | | |
| NXRSWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRSWQ | 300_CHECK_DATE | 4 | | | | 2 | | | | 2 | | | | 576 | | | |
| NXRSWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRSWQ | 400_SPAWN_PROGRAMS | 7 | | | | 2 | | | | 2 | | | | 252 | | | |
| NXRSWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRSWQ | 410_R_SPAWN_PN | 9 | | | | 1 | | | | 1 | | | | 18225 | | | |
| NXRSWQ | 410_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRSWQ | 420_R_SPAWN_PP | 9 | | | | 1 | | | | 1 | | | | 18225 | | | |
| NXRSWQ | 420_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRSWQ | 500_UPDATE_RS_CONTROL_REC | 28 | | | | 4 | | | | 2 | | | | 383292 | | | |
| NXRSWQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRSWQ | 700_CHECK_R_SPAWN_ERR | 5 | | | | 4 | | | | 2 | | | | 180 | | | |
| NXRSWQ | 700_CHECK_R_SPAWN_ERR_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRSWQ | 710_RWAIT | 3 | | | | 1 | | | | 1 | | | | 432 | | | |
| NXRSWQ | 710_RWAIT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRSWQ | 700_STARS_CALL_ERROR | 20 | | | | 6 | | | | 2 | | | | 0 | | | |
| NXRSWQ | 700_STARS_CALL_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRSWQ | 700_SECOND_RSPAWN_ERROR | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRSWQ | 700_EXIT_SECOND_RSPAWN_ERROR | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRSWQ | 700_SPAWN_ERROR | 24 | | | | 8 | | | | 2 | | | | 0 | | | |
| NXRSWQ | 700_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 000_DRIVER | 4 | | | | 2 | | | | 1 | | | | 36 | | | |
| NXRUWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 100_HSKP | 18 | | | | 4 | | | | 1 | | | | 551250 | | | |
| NXRUWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 110_SELECT | 48 | | | | 23 | | | | 2 | | | | 559872 | | | |
| NXRUWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 111_FORMAT | 8 | | | | 2 | | | | 1 | | | | 64800 | | | |
| NXRUWQ | 111_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 200_PROCESS | 24 | | | | 10 | | | | 3 | | | | 685464 | | | |
| NXRUWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 210_ACCUMULATE | 18 | | | | 2 | | | | 2 | | | | 73728 | | | |
| NXRUWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 220_WRITE_TAC_TOTALS | 19 | | | | 6 | | | | 2 | | | | 837900 | | | |
| NXRUWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 230_WRITE_APOE_TOTALS | 18 | | | | 7 | | | | 2 | | | | 438048 | | | |
| NXRUWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 240_WRITE_REPORT_TOTALS | 16 | | | | 6 | | | | 2 | | | | 451584 | | | |
| NXRUWQ | 240_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 250_COPY_TEMP_FILE | 39 | | | | 19 | | | | 2 | | | | 351975 | | | |
| NXRUWQ | 250_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 251_ALASKA_COPY | 6 | | | | 3 | | | | 1 | | | | 216 | | | |
| NXRUWQ | 251_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 252_GERMANY_COPY | 6 | | | | 3 | | | | 1 | | | | 216 | | | |
| NXRUWQ | 252_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 253_HAWAII_COPY | 6 | | | | 3 | | | | 1 | | | | 216 | | | |
| NXRUWQ | 253_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 254_HONDURAS_COPY | 6 | | | | 3 | | | | 1 | | | | 216 | | | |
| NXRUWQ | 254_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 255_JAPAN_COPY | 6 | | | | 3 | | | | 1 | | | | 216 | | | |
| NXRUWQ | 255_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 256_KOREA_COPY | 6 | | | | 3 | | | | 1 | | | | 216 | | | |
| NXRUWQ | 256_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 257_PANAMA_COPY | 6 | | | | 3 | | | | 1 | | | | 216 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXRUWQ | 257_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 258_PUERTO_RICO_COPY | 6 | | | | 3 | | | | 1 | | | | 216 | | | |
| NXRUWQ | 258_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 259_GUAM_COPY | 6 | | | | 3 | | | | 1 | | | | 216 | | | |
| NXRUWQ | 259_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 300_CLOSE | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 700_HEADER | 10 | | | | 2 | | | | 2 | | | | 64000 | | | |
| NXRUWQ | 700_HEADER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 700_END_OF_SECTION | 3 | | | | 2 | | | | 1 | | | | 192 | | | |
| NXRUWQ | 700_EOS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRUWQ | 710_SKIP | 3 | | | | 2 | | | | 1 | | | | 48 | | | |
| NXRUWQ | 710_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRWWQ | 000_RW_DRIVER | 17 | | | | 12 | | | | 2 | | | | 198288 | | | |
| NXRWWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRWWQ | 100_HSKP | 10 | | | | 3 | | | | 1 | | | | 1000 | | | |
| NXRWWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRWWQ | 200_PROCESS_MV | 13 | | | | 6 | | | | 2 | | | | 10192 | | | |
| NXRWWQ | 200_CONTINUE | 5 | | | | 2 | | | | 1 | | | | 405 | | | |
| NXRWWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRWWQ | 300_PROCESS_OH | 13 | | | | 6 | | | | 2 | | | | 10192 | | | |
| NXRWWQ | 300_CONTINUE | 5 | | | | 2 | | | | 1 | | | | 405 | | | |
| NXRWWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRWWQ | 400_WRAPUP | 9 | | | | 3 | | | | 2 | | | | 0 | | | |
| NXRWWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 000_DRIVER | 20 | | | | 7 | | | | 2 | | | | 338000 | | | |
| NXRZWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 100_UPDT_HSKP | 66 | | | | 14 | | | | 2 | | | | 60825600 | | | |
| NXRZWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 110_LOAD_MONTH_YEAR_TABLE | 2 | | | | 3 | | | | 1 | | | | 72 | | | |
| NXRZWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 111_LOAD_MONTH_TABLE | 6 | | | | 2 | | | | 1 | | | | 216 | | | |
| NXRZWQ | 111_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 120_STRIP_OLD_MTH_FRM_OLD_TAPE | 50 | | | | 23 | | | | 6 | | | | 2332800 | | | |
| NXRZWQ | 120_CONTINUE | 34 | | | | 11 | | | | 2 | | | | 2228224 | | | |
| NXRZWQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 121_SEARCH_MONTH_TABLE | 11 | | | | 7 | | | | 2 | | | | 22275 | | | |
| NXRZWQ | 121_SEARCH_MONTH_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 122_SEARCH_YEAR_TABLE | 7 | | | | 4 | | | | 2 | | | | 3087 | | | |
| NXRZWQ | 122_SEARCH_YEAR_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 130_OLD_MONTH_INPUT | 55 | | | | 23 | | | | 6 | | | | 6814720 | | | |
| NXRZWQ | 130_CONTINUE | 35 | | | | 11 | | | | 2 | | | | 2293760 | | | |
| NXRZWQ | 130_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 131_SEARCH_MONTH_TABLE | 11 | | | | 7 | | | | 2 | | | | 22275 | | | |
| NXRZWQ | 131_SEARCH_MONTH_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 132_SEARCH_YEAR_TABLE | 5 | | | | 3 | | | | 1 | | | | 720 | | | |
| NXRZWQ | 132_SEARCH_YEAR_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 133_WRITE_TO_PZ3 | 6 | | | | 2 | | | | 1 | | | | 486 | | | |
| NXRZWQ | 133_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 140_WRITE_PZ2_FILE | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| NXRZWQ | 140_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 141_WRITE_PZ2_MONTH | 3 | | | | 2 | | | | 1 | | | | 27 | | | |
| NXRZWQ | 141_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 200_MERGE_COMPARE | 29 | | | | 34 | | | | 2 | | | | 363776 | | | |
| NXRZWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 210_DELETED_TCMD_PZ3 | 6 | | | | 1 | | | | 1 | | | | 600 | | | |
| NXRZWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 220_DELETED_TCMD_TAPE | 6 | | | | 1 | | | | 1 | | | | 600 | | | |
| NXRZWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 230_COPY_PZ3 | 10 | | | | 9 | | | | 2 | | | | 82810 | | | |
| NXRZWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 240_COPY_TAPE | 6 | | | | 1 | | | | 1 | | | | 600 | | | |
| NXRZWQ | 240_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 250_ABORT_REC_WRITE | 8 | | | | 1 | | | | 1 | | | | 6272 | | | |
| NXRZWQ | 250_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 260_ABORT_TAPE_RECORDS | 17 | | | | 20 | | | | 3 | | | | 170000 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXRZWQ | 260_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 261_READ_SORTED | 5 | | | | 2 | | | | 1 | | | | 180 | | | |
| NXRZWQ | 261_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 270_DELETE_REC_WRITE | 6 | | | | 1 | | | | 1 | | | | 600 | | | |
| NXRZWQ | 270_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 280_WRITE_AND_ABORT_RECORDS | 14 | | | | 6 | | | | 2 | | | | 55566 | | | |
| NXRZWQ | 280_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 281_WRITE_PZ3_TO_TAPE | 13 | | | | 8 | | | | 2 | | | | 140608 | | | |
| NXRZWQ | 281_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 300_WRAP_UP | 4 | | | | 1 | | | | 1 | | | | 400 | | | |
| NXRZWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 310_TOTAL_DISPLAYS | 66 | | | | 2 | | | | 1 | | | | 8986626 | | | |
| NXRZWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 700_DISPLAY_COUNTER_TABLE | 25 | | | | 4 | | | | 2 | | | | 6400 | | | |
| NXRZWQ | 700_DISPLAY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 700_SKIP_TO_HEADERS | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 700_SKIP_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 700_READ_PZ5_TEMP_FILE | 6 | | | | 2 | | | | 1 | | | | 3456 | | | |
| NXRZWQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 700_READ_NEW_FILE | 9 | | | | 3 | | | | 2 | | | | 17424 | | | |
| NXRZWQ | 700_READ_NEW_FILE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 700_WRITE_HISTORY | 4 | | | | 3 | | | | 1 | | | | 3600 | | | |
| NXRZWQ | 700_WRITE_HISTORY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 700_ERROR_DISPLAY | 3 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 700_ERROR_DISPLAY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 700_PZ3_CALCULATE_NEW_YEAR | 4 | | | | 2 | | | | 2 | | | | 144 | | | |
| NXRZWQ | 700_PZ3_CALCULATE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 700_PZ5_CALCULATE_NEW_YEAR | 4 | | | | 2 | | | | 2 | | | | 144 | | | |
| NXRZWQ | 700_PZ5_CALCULATE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXRZWQ | 700_PZ5_FIX_DATES | 27 | | | | 17 | | | | 4 | | | | 78732 | | | |
| NXRZWQ | 700_PZ5_FIX_DATES_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 000_S1_DRIVER | 7 | | | | 5 | | | | 2 | | | | 1008 | | | |
| NXS1WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 100_HSKP | 28 | | | | 4 | | | | 2 | | | | 2621808 | | | |
| NXS1WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 200_BUILD_REPORT | 11 | | | | 16 | | | | 2 | | | | 70400 | | | |
| NXS1WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 210_PROCESS_REC | 5 | | | | 2 | | | | 2 | | | | 180 | | | |
| NXS1WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 211_READ_TC_RECORD | 9 | | | | 4 | | | | 2 | | | | 225 | | | |
| NXS1WQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 212_ADD_TO_TOTALS | 62 | | | | 31 | | | | 3 | | | | 2008800 | | | |
| NXS1WQ | 212_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 213_READ_S1_AKF | 8 | | | | 3 | | | | 2 | | | | 1800 | | | |
| NXS1WQ | 213_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 220_CHANNEL_CHANGE | 15 | | | | 6 | | | | 2 | | | | 11760 | | | |
| NXS1WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 221_DLA_OUTPUT | 19 | | | | 10 | | | | 3 | | | | 121600 | | | |
| NXS1WQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 222_USAF_OUTPUT | 19 | | | | 10 | | | | 3 | | | | 121600 | | | |
| NXS1WQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 223_NAVY_OUTPUT | 19 | | | | 10 | | | | 3 | | | | 121600 | | | |
| NXS1WQ | 223_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 224_DECA_OUTPUT | 19 | | | | 10 | | | | 3 | | | | 121600 | | | |
| NXS1WQ | 224_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 230_NAF_CHANGE | 13 | | | | 6 | | | | 2 | | | | 10192 | | | |
| NXS1WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 231_DLA_OUTPUT | 21 | | | | 12 | | | | 2 | | | | 124509 | | | |
| NXS1WQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 232_USAF_OUTPUT | 21 | | | | 12 | | | | 2 | | | | 124509 | | | |
| NXS1WQ | 232_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 233_NAVY_OUTPUT | 21 | | | | 12 | | | | 2 | | | | 124509 | | | |
| NXS1WQ | 233_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 234_DECA_OUTPUT | 21 | | | | 12 | | | | 2 | | | | 124509 | | | |
| NXS1WQ | 234_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 240_APOE_CHANGE | 13 | | | | 6 | | | | 2 | | | | 13312 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXS1WQ | 240_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 241_DLA_OUTPUT | 17 | | | | 12 | | | | 2 | | | | 140777 | | | |
| NXS1WQ | 241_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 242_USAF_OUTPUT | 17 | | | | 12 | | | | 2 | | | | 140777 | | | |
| NXS1WQ | 242_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 243_NAVY_OUTPUT | 17 | | | | 12 | | | | 2 | | | | 140777 | | | |
| NXS1WQ | 243_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 244_DECA_OUTPUT | 17 | | | | 12 | | | | 2 | | | | 140777 | | | |
| NXS1WQ | 244_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 300_WRAPUP | 3 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXS1WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 700_WRITE_BLANK_LINE_DLA | 5 | | | | 3 | | | | 2 | | | | 320 | | | |
| NXS1WQ | 700_WRITE_BLANK_DLA_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 700_WRITE_BLANK_LINE_DECA | 5 | | | | 3 | | | | 2 | | | | 320 | | | |
| NXS1WQ | 700_WRITE_BLANK_DECA_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 700_WRITE_BLANK_LINE_USAF | 5 | | | | 3 | | | | 2 | | | | 320 | | | |
| NXS1WQ | 700_WRITE_BLANK_USAF_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 700_WRITE_BLANK_LINE_NAVY | 5 | | | | 3 | | | | 2 | | | | 320 | | | |
| NXS1WQ | 700_WRITE_BLANK_NAVY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 700_WRITE_HEADERS_DLA | 17 | | | | 10 | | | | 2 | | | | 45968 | | | |
| NXS1WQ | 700_WRITE_HEADERS_DLA_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 700_WRITE_HEADERS_USAF | 17 | | | | 10 | | | | 2 | | | | 32912 | | | |
| NXS1WQ | 700_WRITE_HEADERS_USAF_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 700_WRITE_HEADERS_NAVY | 17 | | | | 10 | | | | 2 | | | | 32912 | | | |
| NXS1WQ | 700_WRITE_HEADERS_NAVY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 700_WRITE_HEADERS_DECA | 17 | | | | 10 | | | | 2 | | | | 45968 | | | |
| NXS1WQ | 700_WRITE_HEADERS_DECA_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 700_FORMAT_OUTPUT | 38 | | | | 2 | | | | 2 | | | | 78796800 | | | |
| NXS1WQ | 700_FORMAT_OUTPUT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS1WQ | 700_ERROR_ROUTINE | 5 | | | | 2 | | | | 2 | | | | 320 | | | |
| NXS1WQ | 700_ERROR_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 000_DRIVER | 15 | | | | 7 | | | | 2 | | | | 61440 | | | |
| NXS2WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 100_HSKP | 63 | | | | 14 | | | | 2 | | | | 344962800 | | | |
| NXS2WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 110_GETT_AS_OF_DATE | 18 | | | | 5 | | | | 2 | | | | 368082 | | | |
| NXS2WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 120_REFORMAT_SP_A_FILE | 61 | | | | 33 | | | | 2 | | | | 7134804 | | | |
| NXS2WQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 200_BUILD_REPORT | 18 | | | | 9 | | | | 3 | | | | 421362 | | | |
| NXS2WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 210_PROCESS_REC | 59 | | | | 10 | | | | 2 | | | | 585427500 | | | |
| NXS2WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 220_WRITE_CHANNEL_TOTALS | 85 | | | | 4 | | | | 2 | | | | 243343440 | | | |
| NXS2WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 230_CONTROL_SPACE_A_DATA | 13 | | | | 21 | | | | 2 | | | | 67392 | | | |
| NXS2WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 240_WRITE_THEATER_TOTALS | 61 | | | | 7 | | | | 3 | | | | 30061044 | | | |
| NXS2WQ | 240_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 250_WRITE_MFST_STA_TOTALS | 53 | | | | 7 | | | | 3 | | | | 19850832 | | | |
| NXS2WQ | 250_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 300_WRAPUP | 5 | | | | 2 | | | | 2 | | | | 2000 | | | |
| NXS2WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_READ_GEN_REC | 15 | | | | 5 | | | | 2 | | | | 73500 | | | |
| NXS2WQ | 700_READ_GEN_REC_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_PROCESS_SP_A_RECS | 12 | | | | 8 | | | | 3 | | | | 21168 | | | |
| NXS2WQ | 700_PROCESS_SP_A_RECS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_ADD_THTR_SP_A_DATA | 12 | | | | 2 | | | | 2 | | | | 43200 | | | |
| NXS2WQ | 700_ADD_THTR_SP_A_DATA_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_ADD_STA_SP_A_DATA | 12 | | | | 2 | | | | 2 | | | | 15552 | | | |
| NXS2WQ | 700_ADD_STA_SP_A_DATA_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_FORMAT_OUTPUT | 33 | | | | 7 | | | | 2 | | | | 23284800 | | | |
| NXS2WQ | 700_FORMAT_OUTPUT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_ARMY_OUTPUT | 26 | | | | 12 | | | | 2 | | | | 539136 | | | |
| NXS2WQ | 700_ARMY_OUTPUT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_ARMY_PAGE_FILLER | 4 | | | | 3 | | | | 2 | | | | 400 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXS2WQ | 700_ARMY_PAGE_FILLER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_DLA_OUTPUT | 26 | | | | 12 | | | | 2 | | | | 539136 | | | |
| NXS2WQ | 700_DLA_OUTPUT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_DLA_PAGE_FILLER | 4 | | | | 3 | | | | 2 | | | | 400 | | | |
| NXS2WQ | 700_DLA_PAGE_FILLER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_DECA_OUTPUT | 26 | | | | 12 | | | | 2 | | | | 281216 | | | |
| NXS2WQ | 700_DECA_OUTPUT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_DECA_PAGE_FILLER | 4 | | | | 3 | | | | 2 | | | | 400 | | | |
| NXS2WQ | 700_DECA_PAGE_FILLER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_USAF_OUTPUT | 26 | | | | 12 | | | | 2 | | | | 539136 | | | |
| NXS2WQ | 700_USAF_OUTPUT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_USAF_PAGE_FILLER | 4 | | | | 3 | | | | 2 | | | | 400 | | | |
| NXS2WQ | 700_USAF_PAGE_FILLER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_USMC_OUTPUT | 26 | | | | 12 | | | | 2 | | | | 539136 | | | |
| NXS2WQ | 700_USMC_OUTPUT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_USMC_PAGE_FILLER | 4 | | | | 3 | | | | 2 | | | | 400 | | | |
| NXS2WQ | 700_USMC_PAGE_FILLER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_NAVY_OUTPUT | 26 | | | | 12 | | | | 2 | | | | 539136 | | | |
| NXS2WQ | 700_NAVY_OUTPUT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_NAVY_PAGE_FILLER | 4 | | | | 3 | | | | 2 | | | | 400 | | | |
| NXS2WQ | 700_NAVY_PAGE_FILLER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_AMC_OUTPUT | 18 | | | | 9 | | | | 2 | | | | 839808 | | | |
| NXS2WQ | 700_AMC_OUTPUT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_WRITE_AMC_TRAILER | 7 | | | | 3 | | | | 2 | | | | 6300 | | | |
| NXS2WQ | 700_WRITE_AMC_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_WRITE_AMC_HEADERS | 12 | | | | 7 | | | | 2 | | | | 5292 | | | |
| NXS2WQ | 700_WRITE_AMC_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS2WQ | 700_ERROR_ROUTINE | 7 | | | | 3 | | | | 2 | | | | 252 | | | |
| NXS2WQ | 700_ERROR_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 000_S3_DRIVER | 8 | | | | 5 | | | | 2 | | | | 1800 | | | |
| NXS3WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 100_HSKP | 29 | | | | 4 | | | | 2 | | | | 2680064 | | | |
| NXS3WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 200_BUILD_REPORT | 9 | | | | 10 | | | | 2 | | | | 17424 | | | |
| NXS3WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 210_PROCESS_REC | 7 | | | | 4 | | | | 2 | | | | 567 | | | |
| NXS3WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 211_READ_AND_ADD_ONHD_DATA | 18 | | | | 7 | | | | 2 | | | | 43218 | | | |
| NXS3WQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 212_READ_AND_ADD_MVMT_DATA | 19 | | | | 7 | | | | 2 | | | | 59584 | | | |
| NXS3WQ | 212_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 213_READ_S3_AKF | 10 | | | | 6 | | | | 2 | | | | 1440 | | | |
| NXS3WQ | 213_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 220_CHANNEL_CHANGE | 14 | | | | 6 | | | | 2 | | | | 10976 | | | |
| NXS3WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 221_DLA_OUTPUT | 14 | | | | 5 | | | | 3 | | | | 57344 | | | |
| NXS3WQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 222_USAF_OUTPUT | 14 | | | | 5 | | | | 3 | | | | 57344 | | | |
| NXS3WQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 223_NAVY_OUTPUT | 14 | | | | 5 | | | | 3 | | | | 57344 | | | |
| NXS3WQ | 223_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 224_DECA_OUTPUT | 14 | | | | 5 | | | | 3 | | | | 57344 | | | |
| NXS3WQ | 224_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 230_MFST_STA_CHANGE | 13 | | | | 6 | | | | 2 | | | | 10192 | | | |
| NXS3WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 231_DLA_OUTPUT | 11 | | | | 5 | | | | 2 | | | | 9900 | | | |
| NXS3WQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 232_USAF_OUTPUT | 11 | | | | 5 | | | | 2 | | | | 9900 | | | |
| NXS3WQ | 232_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 233_NAVY_OUTPUT | 11 | | | | 5 | | | | 2 | | | | 9900 | | | |
| NXS3WQ | 233_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 234_DECA_OUTPUT | 11 | | | | 5 | | | | 2 | | | | 9900 | | | |
| NXS3WQ | 234_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 300_WRAPUP | 3 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXS3WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 700_WRITE_HEADERS_DLA | 15 | | | | 10 | | | | 2 | | | | 10935 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXS3WQ | 700_WRITE_HEADERS_DLA_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 700_WRITE_HEADERS_USAF | 15 | | | | 10 | | | | 2 | | | | 10935 | | | |
| NXS3WQ | 700_WRITE_HEADERS_USAF_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 700_WRITE_HEADERS_NAVY | 15 | | | | 10 | | | | 2 | | | | 10935 | | | |
| NXS3WQ | 700_WRITE_HEADERS_NAVY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 700_WRITE_HEADERS_DECA | 15 | | | | 10 | | | | 2 | | | | 10935 | | | |
| NXS3WQ | 700_WRITE_HEADERS_DECA_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 700_FORMAT_OUTPUT | 10 | | | | 2 | | | | 2 | | | | 77440 | | | |
| NXS3WQ | 700_FORMAT_OUTPUT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXS3WQ | 700_ERROR_ROUTINE | 5 | | | | 2 | | | | 2 | | | | 320 | | | |
| NXS3WQ | 700_ERROR_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT0WQ | 000_MAIN_DRIVER | 11 | | | | 6 | | | | 2 | | | | 1584 | | | |
| NXT0WQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT0WQ | 100_HSKP | 20 | | | | 4 | | | | 2 | | | | 162000 | | | |
| NXT0WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT0WQ | 200_PROGRAM_SELECT | 12 | | | | 3 | | | | 3 | | | | 5292 | | | |
| NXT0WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT0WQ | 300_WRAPUP | 25 | | | | 3 | | | | 2 | | | | 270400 | | | |
| NXT0WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT0WQ | 210_PROCESS_INPUT | 9 | | | | 5 | | | | 2 | | | | 5184 | | | |
| NXT0WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT0WQ | 700_ERROR_PROCESS | 5 | | | | 1 | | | | 1 | | | | 500 | | | |
| NXT0WQ | 700_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT1WQ | 000_MAIN_DRIVER | 4 | | | | 3 | | | | 1 | | | | 64 | | | |
| NXT1WQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT1WQ | 100_CONTROL | 9 | | | | 6 | | | | 2 | | | | 3600 | | | |
| NXT1WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT1WQ | 110_O_CALL | 16 | | | | 2 | | | | 2 | | | | 102400 | | | |
| NXT1WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT1WQ | 120_I_CALL | 22 | | | | 6 | | | | 2 | | | | 140800 | | | |
| NXT1WQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT1WQ | 130_UPDATE | 14 | | | | 5 | | | | 2 | | | | 24696 | | | |
| NXT1WQ | 130_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT1WQ | 131_ADD | 17 | | | | 2 | | | | 2 | | | | 166617 | | | |
| NXT1WQ | 131_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT1WQ | 132_CHANGE | 32 | | | | 5 | | | | 2 | | | | 903168 | | | |
| NXT1WQ | 132_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT1WQ | 133_DELETE | 18 | | | | 4 | | | | 2 | | | | 139392 | | | |
| NXT1WQ | 133_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT2WQ | 000_MAIN_DRIVER | 3 | | | | 3 | | | | 1 | | | | 48 | | | |
| NXT2WQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT2WQ | 100_CONTROL | 9 | | | | 6 | | | | 2 | | | | 3600 | | | |
| NXT2WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT2WQ | 110_O_CALL | 16 | | | | 2 | | | | 2 | | | | 102400 | | | |
| NXT2WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT2WQ | 120_I_CALL | 22 | | | | 6 | | | | 2 | | | | 140800 | | | |
| NXT2WQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT2WQ | 130_UPDATE | 11 | | | | 4 | | | | 2 | | | | 6875 | | | |
| NXT2WQ | 130_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT2WQ | 131_ADD | 16 | | | | 2 | | | | 2 | | | | 102400 | | | |
| NXT2WQ | 131_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT2WQ | 132_DELETE | 17 | | | | 4 | | | | 2 | | | | 83300 | | | |
| NXT2WQ | 132_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT3WQ | 000_MAIN_DRIVER | 3 | | | | 3 | | | | 1 | | | | 48 | | | |
| NXT3WQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT3WQ | 100_CONTROL | 9 | | | | 6 | | | | 2 | | | | 3600 | | | |
| NXT3WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT3WQ | 110_O_CALL | 16 | | | | 2 | | | | 2 | | | | 102400 | | | |
| NXT3WQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT3WQ | 120_I_CALL | 22 | | | | 6 | | | | 2 | | | | 140800 | | | |
| NXT3WQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT3WQ | 130_UPDATE | 15 | | | | 5 | | | | 2 | | | | 47040 | | | |
| NXT3WQ | 130_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT3WQ | 131_ADD | 17 | | | | 2 | | | | 2 | | | | 166617 | | | |
| NXT3WQ | 131_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT3WQ | 132_CHANGE | 32 | | | | 5 | | | | 2 | | | | 903168 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXT3WQ | 132_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXT3WQ | 133_DELETE | 18 | | | | 4 | | | | 2 | | | | 139392 | | | |
| NXT3WQ | 133_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXTYWQ | 000_MAIN_DRIVER | 6 | | | | 2 | | | | 2 | | | | 24 | | | |
| NXTYWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXTYWQ | 100_SEND_MENU | 15 | | | | 2 | | | | 2 | | | | 96000 | | | |
| NXTYWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXTYWQ | 200_I_CALL | 14 | | | | 5 | | | | 2 | | | | 28350 | | | |
| NXTYWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXTZWQ | 000_DRIVER | 8 | | | | 3 | | | | 2 | | | | 72 | | | |
| NXTZWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXTZWQ | 100_RSPAWN | 26 | | | | 5 | | | | 2 | | | | 103194 | | | |
| NXTZWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXTZWQ | 200_O_CALL | 16 | | | | 2 | | | | 2 | | | | 156816 | | | |
| NXTZWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXTZWQ | 300_I_CALL | 12 | | | | 3 | | | | 2 | | | | 19200 | | | |
| NXTZWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU1WQ | 000_L1_KEY_BUILD_DRIVER | 28 | 28 | | | 10 | 10 | | | 2 | 2 | | | 2086812 | 2086812 | | |
| NXU1WQ | 000_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 100_PROCESS_DATA_DRIVER | 100 | 100 | | | 18 | 18 | | | 3 | 3 | | | 116208400 | 116208400 | | |
| NXU1WQ | 100_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 105_LOAD_PGM_TABLES | 22 | 22 | | | 11 | 11 | | | 3 | 3 | | | 371800 | 371800 | | |
| NXU1WQ | 105_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 110_PROCESS_MOVEMENT | 58 | 58 | | | 21 | 21 | | | 2 | 2 | | | 28096128 | 28096128 | | |
| NXU1WQ | 110_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 111_READ_MOVEMENT | 12 | 12 | | | 4 | 4 | | | 2 | 2 | | | 15552 | 15552 | | |
| NXU1WQ | 111_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 112_SELECT_LG_DELETED_TCMD | 16 | 16 | | | 5 | 5 | | | 2 | 2 | | | 129600 | 129600 | | |
| NXU1WQ | 112_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 115_SELECT_L9_MOVEMENT_CONTROL | 6 | 6 | | | 6 | 6 | | | 2 | 2 | | | 0 | 0 | | |
| NXU1WQ | 1151_FORMAT_L9_MOVEMENT | 15 | 15 | | | 5 | 5 | | | 3 | 3 | | | 77760 | 77760 | | |
| NXU1WQ | 115_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 117_SELECT_LB_MOVEMENT_CONTROL | 14 | 14 | | | 13 | 13 | | | 2 | 2 | | | 0 | 0 | | |
| NXU1WQ | 1171_FORMAT_LB_MOVEMENT | 15 | 15 | | | 3 | 3 | | | 2 | 2 | | | 365040 | 365040 | | |
| NXU1WQ | 117_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 118_SELECT_LE_MOVEMENT_CONTROL | 8 | 8 | | | 5 | 5 | | | 2 | 2 | | | 0 | 0 | | |
| NXU1WQ | 1181_FORMAT_LE_PALLET_MOVEMENT | 13 | 13 | | | 3 | 3 | | | 2 | 2 | | | 130000 | 130000 | | |
| NXU1WQ | 118_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 119_SELECT_LF_MOVEMENT_CONTROL | 12 | 12 | | | 8 | 8 | | | 2 | 2 | | | 0 | 0 | | |
| NXU1WQ | 1191_FORMAT_LF_MOVEMENT | 13 | 13 | | | 3 | 3 | | | 2 | 2 | | | 63700 | 63700 | | |
| NXU1WQ | 119_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 11A_SELECT_LH_MOVEMENT_CONTROL | 8 | 8 | | | 6 | 6 | | | 2 | 2 | | | 288 | 288 | | |
| NXU1WQ | 11A1_FORMAT_LH_PALLET_MOVEMENT | 10 | 10 | | | 3 | 3 | | | 2 | 2 | | | 24010 | 24010 | | |
| NXU1WQ | 11A_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 11C_SELECT_S3_MOVEMENT_CONTROL | 14 | 14 | | | 16 | 16 | | | 2 | 2 | | | 0 | 0 | | |
| NXU1WQ | 11C1_FORMAT_S3_MOVEMENT | 12 | 12 | | | 3 | 3 | | | 2 | 2 | | | 34992 | 34992 | | |
| NXU1WQ | 11C_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 11E_SELECT_LP_MOVEMENT_CONTROL | 14 | 14 | | | 12 | 12 | | | 3 | 3 | | | 0 | 0 | | |
| NXU1WQ | 11E1_FORMAT_LP_MOVEMENT | 7 | 7 | | | 2 | 2 | | | 1 | 1 | | | 9072 | 9072 | | |
| NXU1WQ | 11E_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 120_PROCESS_ON_HAND | 37 | 37 | | | 16 | 16 | | | 2 | 2 | | | 4795200 | 4795200 | | |
| NXU1WQ | 120_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 121_READ_ON_HAND | 11 | 11 | | | 4 | 4 | | | 2 | 2 | | | 9900 | 9900 | | |
| NXU1WQ | 121_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 128_FORMAT_LF_ON_HAND | 21 | 21 | | | 12 | 12 | | | 2 | 2 | | | 302400 | 302400 | | |
| NXU1WQ | 128_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 129_FORMAT_S3_ON_HAND | 20 | 20 | | | 22 | 22 | | | 2 | 2 | | | 233280 | 233280 | | |
| NXU1WQ | 129_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 200_LOAD_FILES_DRIVER | 21 | 21 | | | 11 | 11 | | | 2 | 2 | | | 134400 | 134400 | | |
| NXU1WQ | 200_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 210_ALLOCATE_AKF_FILE | 19 | 29 | | | 11 | 14 | | | 2 | 2 | | | 134064 | 1160000 | | |
| NXU1WQ | 210_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 220_AKF_FILE_BUILD | 8 | 8 | | | 5 | 5 | | | 2 | 2 | | | 1152 | 1152 | | |
| NXU1WQ | 220_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 230_RELEASE_AKF_FILE | 14 | 14 | | | 5 | 5 | | | 2 | 2 | | | 113400 | 113400 | | |
| NXU1WQ | 230_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXU1WQ | 240_AKF_CLEANUP | 20 | 20 | | | 10 | 10 | | | 2 | 2 | | | 327680 | 327680 | | |
| NXU1WQ | 240_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 300_REBUILD_PROCESS | 21 | 21 | | | 6 | 6 | | | 2 | 2 | | | 18900 | 18900 | | |
| NXU1WQ | 300_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 310_FIND_RC | 5 | 5 | | | 3 | 3 | | | 2 | 2 | | | 45 | 45 | | |
| NXU1WQ | 310_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 320_LOAD_TEMP_AKF | 8 | 8 | | | 4 | 4 | | | 2 | 2 | | | 512 | 512 | | |
| NXU1WQ | 320_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 700_ADDFIL_ERROR_CHECK | 12 | 12 | | | 4 | 4 | | | 2 | 2 | | | 43200 | 43200 | | |
| NXU1WQ | 700_ADDFIL_ERROR_CHECK_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 700_SEARCH_PGM_STA_TABLE | 6 | 6 | | | 6 | 6 | | | 2 | 2 | | | 1176 | 1176 | | |
| NXU1WQ | 700_SEARCH_PGM_STA_TABLE_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 700_SEARCH_CHNL_TABLE_DTL | 11 | 11 | | | 7 | 7 | | | 2 | 2 | | | 14256 | 14256 | | |
| NXU1WQ | 700_SEARCH_CHNL_TABLE_DTL_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 700_SEARCH_CHNL_TABLE_PLT | 11 | 11 | | | 7 | 7 | | | 2 | 2 | | | 14256 | 14256 | | |
| NXU1WQ | 700_SEARCH_CHNL_TABLE_PLT_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 700_CHECK_NAF_INDICATORS | 18 | 18 | | | 5 | 5 | | | 2 | 2 | | | 10368 | 10368 | | |
| NXU1WQ | 700_CHECK_NAF_INDICATORS_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU1WQ | 700_FORMAT_NAF | 24 | 24 | | | 25 | 25 | | | 2 | 2 | | | 10584 | 10584 | | |
| NXU1WQ | 700_FORMAT_NAF_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| NXU3WQ | 000_U3_DRIVER | 24 | | | | 11 | | | | 2 | | | | 345600 | | | |
| NXU3WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 100_HSKP | 27 | | | | 4 | | | | 2 | | | | 2193075 | | | |
| NXU3WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 200_BUILD_REPORT | 9 | | | | 10 | | | | 2 | | | | 17424 | | | |
| NXU3WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 210_PROCESS_REC | 7 | | | | 4 | | | | 2 | | | | 567 | | | |
| NXU3WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 211_READ_AND_ADD_ONHD_DATA | 18 | | | | 7 | | | | 2 | | | | 43218 | | | |
| NXU3WQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 212_READ_AND_ADD_MVMT_DATA | 19 | | | | 7 | | | | 2 | | | | 59584 | | | |
| NXU3WQ | 212_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 213_READ_U3_AKF | 10 | | | | 6 | | | | 2 | | | | 1440 | | | |
| NXU3WQ | 213_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 215_NO_DATA | 10 | | | | 6 | | | | 2 | | | | 5760 | | | |
| NXU3WQ | 215_NO_DATA_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 220_CHANNEL_CHANGE | 14 | | | | 6 | | | | 2 | | | | 10976 | | | |
| NXU3WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 221_DLA_OUTPUT | 14 | | | | 5 | | | | 3 | | | | 57344 | | | |
| NXU3WQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 222_USAF_OUTPUT | 14 | | | | 5 | | | | 3 | | | | 57344 | | | |
| NXU3WQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 223_NAVY_OUTPUT | 14 | | | | 5 | | | | 3 | | | | 57344 | | | |
| NXU3WQ | 223_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 224_DECA_OUTPUT | 14 | | | | 5 | | | | 3 | | | | 57344 | | | |
| NXU3WQ | 224_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 230_MFST_STA_CHANGE | 13 | | | | 6 | | | | 2 | | | | 10192 | | | |
| NXU3WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 231_DLA_OUTPUT | 11 | | | | 5 | | | | 2 | | | | 9900 | | | |
| NXU3WQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 232_USAF_OUTPUT | 11 | | | | 5 | | | | 2 | | | | 9900 | | | |
| NXU3WQ | 232_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 233_NAVY_OUTPUT | 10 | | | | 4 | | | | 2 | | | | 6250 | | | |
| NXU3WQ | 233_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 234_DECA_OUTPUT | 11 | | | | 4 | | | | 2 | | | | 6875 | | | |
| NXU3WQ | 234_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 300_WRAPUP | 3 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXU3WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 700_WRITE_HEADERS_DLA | 15 | | | | 10 | | | | 2 | | | | 16335 | | | |
| NXU3WQ | 700_WRITE_HEADERS_DLA_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 700_WRITE_HEADERS_DECA | 15 | | | | 10 | | | | 2 | | | | 16335 | | | |
| NXU3WQ | 700_WRITE_HEADERS_DECA_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 700_WRITE_HEADERS_USAF | 15 | | | | 10 | | | | 2 | | | | 16335 | | | |
| NXU3WQ | 700_WRITE_HEADERS_USAF_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 700_WRITE_HEADERS_NAVY | 15 | | | | 10 | | | | 2 | | | | 16335 | | | |
| NXU3WQ | 700_WRITE_HEADERS_NAVY_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXU3WQ | 700_FORMAT_OUTPUT | 10 | | | | 2 | | | | 2 | | | | 77440 | | | |
| NXU3WQ | 700_FORMAT_OUTPUT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU3WQ | 700_ERROR_ROUTINE | 5 | | | | 2 | | | | 2 | | | | 320 | | | |
| NXU3WQ | 700_ERROR_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 000_U9_CONTROL_DRIVER | 15 | | | | 7 | | | | 2 | | | | 26460 | | | |
| NXU9WQ | 000_DRIVER_COMPLETE | 6 | | | | 3 | | | | 2 | | | | 294 | | | |
| NXU9WQ | 100_HSKP_DRIVER | 33 | | | | 7 | | | | 2 | | | | 8250000 | | | |
| NXU9WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 200_UPDATE_DRIVER | 37 | | | | 29 | | | | 4 | | | | 447700 | | | |
| NXU9WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 210_CREATE | 7 | | | | 5 | | | | 2 | | | | 5488 | | | |
| NXU9WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 215_NO_DATA | 5 | | | | 2 | | | | 2 | | | | 80 | | | |
| NXU9WQ | 215_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 220_COPY | 5 | | | | 4 | | | | 2 | | | | 1620 | | | |
| NXU9WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 230_UPDATE | 39 | | | | 21 | | | | 3 | | | | 5054400 | | | |
| NXU9WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 231_SUPER_PRIORITY_ADD | 6 | | | | 2 | | | | 2 | | | | 2400 | | | |
| NXU9WQ | 231_ADD_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 231_SUPER_PRIORITY_SUBTRACT | 10 | | | | 2 | | | | 2 | | | | 16000 | | | |
| NXU9WQ | 231_SUBT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 232_PRIORITY_1_2_3_ADD | 6 | | | | 2 | | | | 2 | | | | 2400 | | | |
| NXU9WQ | 232_ADD_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 232_PRIORITY_1_2_3_SUBTRACT | 10 | | | | 2 | | | | 2 | | | | 16000 | | | |
| NXU9WQ | 232_SUBT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 233_SUPER_THRU_3_ADD | 7 | | | | 2 | | | | 2 | | | | 5488 | | | |
| NXU9WQ | 233_ADD_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 233_SUPER_THRU_3_SUBTRACT | 11 | | | | 2 | | | | 2 | | | | 34496 | | | |
| NXU9WQ | 233_SUBT_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| NXU9WQ | 234_PRIORITY_4_ADD | 6 | | | | 2 | | | | 2 | | | | 2400 | | | |
| NXU9WQ | 234_ADD_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 234_PRIORITY_4_SUBTRACT | 10 | | | | 2 | | | | 2 | | | | 16000 | | | |
| NXU9WQ | 234_SUBT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 300_BUILD_REPORT_DRIVER | 35 | | | | 8 | | | | 2 | | | | 2551500 | | | |
| NXU9WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 310_TYPE_PROCESS | 86 | | | | 7 | | | | 3 | | | | 267605856 | | | |
| NXU9WQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 311_READ_NEW_SUBT | 6 | | | | 3 | | | | 2 | | | | 864 | | | |
| NXU9WQ | 311_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 320_SUB_TOTAL_OUTPUT | 62 | | | | 8 | | | | 2 | | | | 57139200 | | | |
| NXU9WQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 330_STN_TOTAL_OUTPUT | 41 | | | | 5 | | | | 2 | | | | 12492864 | | | |
| NXU9WQ | 330_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 400_WRAP_UP | 5 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXU9WQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 700_WRITE_TRAILER | 7 | | | | 3 | | | | 2 | | | | 17500 | | | |
| NXU9WQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 700_WRITE_HEADA | 4 | | | | 3 | | | | 2 | | | | 1024 | | | |
| NXU9WQ | 700_WRITE_HEADA_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 700_WRITE_SUB_HEADINGS | 18 | | | | 11 | | | | 2 | | | | 101250 | | | |
| NXU9WQ | 700_WRITE_SUB_HEADINGS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXU9WQ | 700_NO_DATA_RPT | 10 | | | | 8 | | | | 2 | | | | 810 | | | |
| NXU9WQ | 700_NO_DATA_RPT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 000_DRIVER | 22 | | | | 10 | | | | 2 | | | | 495000 | | | |
| NXUBWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 100_HOUSEKEEPING | 50 | | | | 11 | | | | 3 | | | | 79380000 | | | |
| NXUBWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 110_LOAD_MAI_TABLE | 6 | | | | 2 | | | | 1 | | | | 1350 | | | |
| NXUBWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 200_UPDATE_DRIVER | 6 | | | | 4 | | | | 2 | | | | 384 | | | |
| NXUBWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 210_READ_CONTROL | 8 | | | | 5 | | | | 2 | | | | 1800 | | | |
| NXUBWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 211_READ_AKF | 7 | | | | 3 | | | | 2 | | | | 1575 | | | |
| NXUBWQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXUBWQ | 212_READ_DAILY_MOVEMENT | 12 | | | | 4 | | | | 2 | | | | 9408 | | | |
| NXUBWQ | 212_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 213_READ_OLD_CUM_MVMT | 10 | | | | 3 | | | | 2 | | | | 12960 | | | |
| NXUBWQ | 213_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 215_NO_DATA | 4 | | | | 2 | | | | 2 | | | | 36 | | | |
| NXUBWQ | 215_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 220_UPDATE_AND_WRITE_CONTROL | 14 | | | | 6 | | | | 2 | | | | 12600 | | | |
| NXUBWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 221_WRITE_CUM_AND_DAILY_REC | 12 | | | | 4 | | | | 2 | | | | 37632 | | | |
| NXUBWQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 222_CREATE_NEW_CUM_REC | 87 | | | | 47 | | | | 3 | | | | 74278512 | | | |
| NXUBWQ | 222_CONTINUE | 16 | | | | 6 | | | | 2 | | | | 46656 | | | |
| NXUBWQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 223_UPDATE_CUM_REC | 156 | | | | 54 | | | | 4 | | | | 215744256 | | | |
| NXUBWQ | 223_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 300_REPORT_DRIVER | 6 | | | | 3 | | | | 2 | | | | 486 | | | |
| NXUBWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 310_READ_DAILY_AND_CUM | 11 | | | | 4 | | | | 2 | | | | 26411 | | | |
| NXUBWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 320_FORMAT_AND_WRITE_CONTROL | 12 | | | | 9 | | | | 2 | | | | 24300 | | | |
| NXUBWQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 321_SUB_AREA_PROCESS | 14 | | | | 4 | | | | 2 | | | | 68600 | | | |
| NXUBWQ | 321_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 322_MAI_PROCESS | 16 | | | | 5 | | | | 2 | | | | 123904 | | | |
| NXUBWQ | 322_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 323_STATION_PROCESS | 96 | | | | 11 | | | | 2 | | | | 3538944 | | | |
| NXUBWQ | 323_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 324_APOD_PROCESS | 53 | | | | 15 | | | | 4 | | | | 2588573 | | | |
| NXUBWQ | 324_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 330_WRITE_GRAND_TOTALS | 5 | | | | 2 | | | | 2 | | | | 180 | | | |
| NXUBWQ | 330_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 331_WRITE_21ST_AF_TOTALS | 44 | | | | 7 | | | | 2 | | | | 228096 | | | |
| NXUBWQ | 331_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 332_WRITE_22ND_AF_TOTALS | 44 | | | | 7 | | | | 2 | | | | 228096 | | | |
| NXUBWQ | 332_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 333_WRITE_AMC_HQ_TOTALS | 57 | | | | 7 | | | | 2 | | | | 295488 | | | |
| NXUBWQ | 333_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 400_WRAPUP | 3 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXUBWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 700_COMPUTE_PROCESS | 191 | | | | 14 | | | | 2 | | | | 3482483900 | | | |
| NXUBWQ | 700_COMPUTE_PROCESS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 700_WRITE_HEADER_PART_1 | 18 | | | | 10 | | | | 3 | | | | 130050 | | | |
| NXUBWQ | 700_WRITE_HEADER_PART1_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 700_WRITE_HEADER_PART_2 | 9 | | | | 8 | | | | 2 | | | | 2916 | | | |
| NXUBWQ | 700_WRITE_HEADER_PART2_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 700_WRITE_AF_TOTAL_HDR | 9 | | | | 8 | | | | 2 | | | | 2916 | | | |
| NXUBWQ | 700_WRITE_AF_TOTAL_HDR_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 700_WRITE_AMC_TOTAL_HDR | 9 | | | | 8 | | | | 2 | | | | 2304 | | | |
| NXUBWQ | 700_WRITE_AMC_TOTAL_HDR_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 700_WRITE_TRAILER | 14 | | | | 5 | | | | 2 | | | | 229376 | | | |
| NXUBWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 700_SEARCH_MAI_CODE | 6 | | | | 4 | | | | 2 | | | | 1176 | | | |
| NXUBWQ | 700_SEARCH_MAI_CODE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUBWQ | 700_NO_DATA_RPT | 12 | | | | 10 | | | | 2 | | | | 972 | | | |
| NXUBWQ | 700_NO_DATA_RPT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 000_LE_DRIVER | 15 | | | | 8 | | | | 3 | | | | 73500 | | | |
| NXUEWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 100_HOUSEKEEPING | 21 | | | | 2 | | | | 1 | | | | 2601984 | | | |
| NXUEWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 200_BUILD_REPORT | 7 | | | | 1 | | | | 1 | | | | 252 | | | |
| NXUEWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 210_READ_DETAIL_AND_NEW_AKF | 28 | | | | 5 | | | | 3 | | | | 3568572 | | | |
| NXUEWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 215_NO_DATA | 4 | | | | 1 | | | | 1 | | | | 16 | | | |
| NXUEWQ | 215_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 220_PROCESS_PRIORITY_DATA | 21 | | | | 25 | | | | 3 | | | | 108864 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXUEWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 230_PROCESS_DETAIL | 21 | | | | 19 | | | | 3 | | | | 571725 | | | |
| NXUEWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 231_WRITE_DETAIL | 36 | | | | 7 | | | | 2 | | | | 33246756 | | | |
| NXUEWQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 240_PROCESS_MANIFEST_DEST | 25 | | | | 17 | | | | 3 | | | | 1440000 | | | |
| NXUEWQ | 240_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 241_WRITE_MANIFEST_DEST | 20 | | | | 4 | | | | 1 | | | | 1479680 | | | |
| NXUEWQ | 241_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 250_PROCESS_MISSION | 27 | | | | 16 | | | | 3 | | | | 1529388 | | | |
| NXUEWQ | 250_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 251_WRITE_PRI1_3_TOTAL | 14 | | | | 2 | | | | 1 | | | | 129024 | | | |
| NXUEWQ | 251_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 252_WRITE_TP4_TOTAL | 19 | | | | 3 | | | | 2 | | | | 393984 | | | |
| NXUEWQ | 252_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 253_WRITE_MISSION | 23 | | | | 5 | | | | 2 | | | | 1907712 | | | |
| NXUEWQ | 253_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 260_PROCESS_STATION | 40 | | | | 17 | | | | 3 | | | | 14113440 | | | |
| NXUEWQ | 260_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 261_WRITE_STATION | 18 | | | | 3 | | | | 1 | | | | 911250 | | | |
| NXUEWQ | 261_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 300_WRAPUP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 700_WRITE_TRAILER | 6 | | | | 2 | | | | 1 | | | | 15000 | | | |
| NXUEWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 700_WRITE_HEADERS | 9 | | | | 7 | | | | 1 | | | | 8100 | | | |
| NXUEWQ | 700_WRITE_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 700_ERROR_ROUTINE | 6 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 700_ERROR_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUEWQ | 700_NO_DATA_RPT | 8 | | | | 8 | | | | 1 | | | | 512 | | | |
| NXUEWQ | 700_NO_DATA_RPT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 000_UF_DRIVER | 15 | | | | 10 | | | | 2 | | | | 96000 | | | |
| NXUFWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 100_HSKP | 63 | | | | 14 | | | | 3 | | | | 50014503 | | | |
| NXUFWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 110_LOAD_STATION_TABLE | 22 | | | | 9 | | | | 3 | | | | 79200 | | | |
| NXUFWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 120_RESEQUENCE_SP_A_FILE | 6 | | | | 3 | | | | 2 | | | | 600 | | | |
| NXUFWQ | 120_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 121_PROCESS_SPACE_A_FILE | 9 | | | | 5 | | | | 2 | | | | 1296 | | | |
| NXUFWQ | 121_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 200_GEN_FILE_UPDATE | 49 | | | | 16 | | | | 4 | | | | 2480625 | | | |
| NXUFWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 210_UPDATE_WITH_TEMP_GEN_REC | 40 | | | | 3 | | | | 2 | | | | 7464960 | | | |
| NXUFWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 215_NO_DATA | 5 | | | | 2 | | | | 2 | | | | 80 | | | |
| NXUFWQ | 215_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 220_UPDATE_WITH_TC_REC | 14 | | | | 8 | | | | 3 | | | | 22400 | | | |
| NXUFWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 221_PROCESS_MVMT_LIFT_24 | 49 | | | | 20 | | | | 8 | | | | 1254400 | | | |
| NXUFWQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 222_PROCESS_MVMT_LIFT | 35 | | | | 20 | | | | 8 | | | | 283500 | | | |
| NXUFWQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 223_PROCESS_MVMT_SUBT | 35 | | | | 20 | | | | 8 | | | | 283500 | | | |
| NXUFWQ | 223_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 224_PROCESS_ONHAND | 35 | | | | 20 | | | | 8 | | | | 283500 | | | |
| NXUFWQ | 224_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 300_BUILD_REPORT | 8 | | | | 6 | | | | 2 | | | | 4608 | | | |
| NXUFWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 310_HSKP | 13 | | | | 3 | | | | 2 | | | | 63700 | | | |
| NXUFWQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 320_BUILD_PORTS_REPORT | 14 | | | | 7 | | | | 2 | | | | 204974 | | | |
| NXUFWQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 321_PROCESS_GEN_REC | 7 | | | | 3 | | | | 2 | | | | 0 | | | |
| NXUFWQ | 321_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 322_PROCESS_SP_ASSIGN_REC | 7 | | | | 4 | | | | 3 | | | | 700 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXUFWQ | 322_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 3221_LOAD_SP_ASSIGN_REC | 51 | | | | 3 | | | | 2 | | | | 249900 | | | |
| NXUFWQ | 3221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 330_WRITE_HQS_TOTALS | 14 | | | | 2 | | | | 2 | | | | 6174 | | | |
| NXUFWQ | 330_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 340_DISPLAY_COMPUTES | 26 | | | | 2 | | | | 2 | | | | 1124864 | | | |
| NXUFWQ | 340_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 400_WRAPUP | 9 | | | | 3 | | | | 2 | | | | 9801 | | | |
| NXUFWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 700_ERROR_ROUTINE | 4 | | | | 2 | | | | 2 | | | | 256 | | | |
| NXUFWQ | 700_ERROR_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 700_SEARCH_STATION_TABLE | 8 | | | | 7 | | | | 2 | | | | 512 | | | |
| NXUFWQ | 700_SEARCH_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 700_READ_UF_AKF | 10 | | | | 5 | | | | 2 | | | | 1440 | | | |
| NXUFWQ | 700_READ_AKF_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 700_READ_TEMP_GEN_REC | 13 | | | | 5 | | | | 2 | | | | 5733 | | | |
| NXUFWQ | 700_READ_TEMP_GEN_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 700_READ_TC_OH_REC | 11 | | | | 4 | | | | 2 | | | | 3564 | | | |
| NXUFWQ | 700_READ_TC_OH_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 700_READ_TC_MV_REC | 12 | | | | 4 | | | | 2 | | | | 6912 | | | |
| NXUFWQ | 700_READ_TC_MV_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 700_READ_GEN_REC | 9 | | | | 4 | | | | 2 | | | | 1296 | | | |
| NXUFWQ | 700_READ_GEN_REC_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 700_READ_SPACE_A_FILE | 7 | | | | 4 | | | | 2 | | | | 448 | | | |
| NXUFWQ | 700_READ_SPACE_A_FILE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 700_WRITE_OUTPUT | 47 | | | | 18 | | | | 2 | | | | 5306112 | | | |
| NXUFWQ | 700_WRITE_OUTPUT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 700_CONVERT_DATA | 47 | | | | 3 | | | | 2 | | | | 114086108 | | | |
| NXUFWQ | 700_CONVERT_DATA_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 700_LOAD_AS_OF_DATE | 15 | | | | 6 | | | | 2 | | | | 181500 | | | |
| NXUFWQ | 700_LOAD_AS_OF_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUFWQ | 700_NO_DATA_RPT | 10 | | | | 9 | | | | 2 | | | | 1000 | | | |
| NXUFWQ | 700_NO_DATA_RPT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUGWQ | 000_DRIVER | 12 | | | | 7 | | | | 4 | | | | 27648 | | | |
| NXUGWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUGWQ | 100_HOUSEKEEPING | 27 | | | | 5 | | | | 3 | | | | 6750000 | | | |
| NXUGWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUGWQ | 200_PROCESS_RECORDS | 15 | | | | 7 | | | | 2 | | | | 73500 | | | |
| NXUGWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUGWQ | 210_PROCESS_NEW_MFST_STATION | 13 | | | | 4 | | | | 2 | | | | 46800 | | | |
| NXUGWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUGWQ | 215_NO_DATA | 6 | | | | 3 | | | | 1 | | | | 486 | | | |
| NXUGWQ | 215_NO_DATA_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUGWQ | 220_WRITE_PART_TWO_HEADING | 13 | | | | 3 | | | | 2 | | | | 16848 | | | |
| NXUGWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUGWQ | 230_PRODUCE_DETAIL_LINE | 10 | | | | 4 | | | | 2 | | | | 7840 | | | |
| NXUGWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUGWQ | 231_FORMAT_DETAIL_LINE | 39 | | | | 3 | | | | 2 | | | | 35942400 | | | |
| NXUGWQ | 231_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUGWQ | 300_WRAPUP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUGWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUGWQ | 700_READ_AKF_AND_PZ1 | 11 | | | | 3 | | | | 1 | | | | 6336 | | | |
| NXUGWQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUGWQ | 700_WRITE_TRAILER | 9 | | | | 3 | | | | 2 | | | | 20736 | | | |
| NXUGWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUGWQ | 700_WRITE_HEADINGS | 12 | | | | 8 | | | | 2 | | | | 6912 | | | |
| NXUGWQ | 700_WRITE_HEADINGS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUGWQ | 700_ERROR_ROUTINE | 6 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUGWQ | 700_ERROR_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUGWQ | 700_NO_DATA_HDRS | 9 | | | | 8 | | | | 1 | | | | 576 | | | |
| NXUGWQ | 700_NO_DATA_HDRS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUHWQ | 000_UH_DRIVER | 17 | | | | 9 | | | | 3 | | | | 131648 | | | |
| NXUHWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUHWQ | 100_HSKP | 32 | | | | 6 | | | | 3 | | | | 8820000 | | | |
| NXUHWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUHWQ | 200_PROCESS | 14 | | | | 3 | | | | 2 | | | | 42350 | | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXUHWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUHWQ | 210_PROCESS_DETAILS | 21 | | | | 7 | | | | 2 | | | | 162624 | | | |
| NXUHWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUHWQ | 211_FORMAT_REPORT | 36 | | | | 2 | | | | 2 | | | | 50979600 | | | |
| NXUHWQ | 211_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUHWQ | 212_UPDATE_COUNTERS | 135 | | | | 38 | | | | 2 | | | | 34292160 | | | |
| NXUHWQ | 212_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUHWQ | 215_NO_DATA | 5 | | | | 2 | | | | 2 | | | | 80 | | | |
| NXUHWQ | 215_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUHWQ | 220_PROCESS_TOTALS | 4 | | | | 2 | | | | 2 | | | | 64 | | | |
| NXUHWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUHWQ | 221_FORMAT_TOTALS | 65 | | | | 2 | | | | 2 | | | | 1089986625 | | | |
| NXUHWQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUHWQ | 222_WRITE_TOTALS | 32 | | | | 31 | | | | 2 | | | | 115200 | | | |
| NXUHWQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUHWQ | 300_WRAPUP | 3 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXUHWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUHWQ | 700_WRITE_HEADER_PART_1 | 10 | | | | 8 | | | | 2 | | | | 25000 | | | |
| NXUHWQ | 700_WRITE_HEADER_PART1_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUHWQ | 700_WRITE_HEADER_PART_2 | 7 | | | | 6 | | | | 2 | | | | 1008 | | | |
| NXUHWQ | 700_WRITE_HEADER_PART2_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUHWQ | 700_WRITE_TRAILER | 7 | | | | 3 | | | | 2 | | | | 8575 | | | |
| NXUHWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUHWQ | 700_ERROR_ROUTINE | 5 | | | | 2 | | | | 2 | | | | 0 | | | |
| NXUHWQ | 700_ERROR_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUHWQ | 700_NO_DATA_RPT | 10 | | | | 10 | | | | 2 | | | | 1000 | | | |
| NXUHWQ | 700_NO_DATA_RPT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUPWQ | 000_UP_DRIVER | 15 | | | | 8 | | | | 2 | | | | 54000 | | | |
| NXUPWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUPWQ | 100_HSKP | 19 | | | | 2 | | | | 1 | | | | 1543275 | | | |
| NXUPWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUPWQ | 200_BUILD_REPORT | 18 | | | | 4 | | | | 3 | | | | 373248 | | | |
| NXUPWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUPWQ | 210_READ_UP_AKF | 4 | | | | 2 | | | | 1 | | | | 64 | | | |
| NXUPWQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUPWQ | 215_NO_DATA | 7 | | | | 2 | | | | 1 | | | | 1008 | | | |
| NXUPWQ | 215_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUPWQ | 220_PROCESS_REPORT_DATA | 17 | | | | 6 | | | | 3 | | | | 88128 | | | |
| NXUPWQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUPWQ | 221_WRITE_APOD_LINE | 14 | | | | 3 | | | | 2 | | | | 83006 | | | |
| NXUPWQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUPWQ | 222_WRITE_STATION_TOTALS | 15 | | | | 4 | | | | 2 | | | | 88935 | | | |
| NXUPWQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUPWQ | 230_WRITE_GRAND_TOTALS | 8 | | | | 2 | | | | 1 | | | | 6272 | | | |
| NXUPWQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUPWQ | 300_WRAPUP | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUPWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUPWQ | 700_READ_AND_ADD | 11 | | | | 2 | | | | 1 | | | | 13475 | | | |
| NXUPWQ | 700_READ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUPWQ | 700_WRITE_TRAILER | 6 | | | | 2 | | | | 1 | | | | 9600 | | | |
| NXUPWQ | 700_WRITE_TRAILER_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUPWQ | 700_WRITE_HEADERS | 8 | | | | 6 | | | | 1 | | | | 3200 | | | |
| NXUPWQ | 700_WRITE_HEADERS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUPWQ | 700_ERROR_ROUTINE | 6 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUPWQ | 700_ERROR_ROUTINE_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUXWQ | 000_DRIVER | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUXWQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUXWQ | 100_SPAWN_U0 | 9 | | | | 1 | | | | 1 | | | | 18225 | | | |
| NXUXWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUXWQ | 110_CHECK_R_SPAWN_ERR | 5 | | | | 4 | | | | 2 | | | | 20 | | | |
| NXUXWQ | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NXUXWQ | 111_RWAIT | 7 | | | | 2 | | | | 2 | | | | 9072 | | | |
| NXUXWQ | 111_RWAIT_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| TPFOWQ | 0000_MAIN | 8 | | | | 2 | | | | 2 | | | | 2592 | | | |
| TPFOWQ | 0000_EXIT | 2 | | | | 2 | | | | 2 | | | | 0 | | | |
| TPFOWQ | 0000_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| TPFOWQ | 1000_CHECK_STATIS | 27 | | | | 17 | | | | 6 | | | | 944163 | | | |
| TPFOWQ | 1000_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| TPFOWQ | 2000_VALIDATE_DEVICE | 25 | | | | 14 | | | | 5 | | | | 828100 | | | |
| TPFOWQ | 2000_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| TPFOWQ | 2100_VALIDATE_CRT | 7 | | | | 5 | | | | 3 | | | | 1792 | | | |
| TPFOWQ | 2100_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| TPFOWQ | 2200_VALIDATE_PBF | 14 | | | | 8 | | | | 5 | | | | 33614 | | | |
| TPFOWQ | 2200_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| TPFOWQ | 2300_VALIDATE_ABF | 29 | | | | 17 | | | | 6 | | | | 2145536 | | | |
| TPFOWQ | 2300_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| TPFOWQ | 2400_VALIDATE_MINI | 38 | | | | 23 | | | | 4 | | | | 4522950 | | | |
| TPFOWQ | 2400_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| TPFOWQ | 2500_VALIDATE_PPAL | 4 | | | | 5 | | | | 3 | | | | 144 | | | |
| TPFOWQ | 2500_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| TPFOWQ | 3000_PROCESS_MESSAGE | 26 | | | | 5 | | | | 4 | | | | 7514 | | | |
| TPFOWQ | 3000_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| TPFOWQ | 4000_SETUP_MESSAGE | 8 | | | | 5 | | | | 3 | | | | 10368 | | | |
| TPFOWQ | 4000_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| TPFOWQ | 4010_SETUP_PRINT_MESSAGE | 54 | | | | 6 | | | | 2 | | | | 33106806 | | | |
| TPFOWQ | 4010_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| TPFOWQ | 4100_ELIMINATE_DUPLICATES | 15 | | | | 3 | | | | 2 | | | | 8640 | | | |
| TPFOWQ | 4100_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| TPFOWQ | 4200_VALIDATE_OUT_IDS | 36 | | | | 6 | | | | 3 | | | | 298116 | | | |
| TPFOWQ | 4200_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| TPFOWQ | 4300_STORE_DTM_RECORD | 37 | | | | 3 | | | | 2 | | | | 15632500 | | | |
| TPFOWQ | 4300_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| TPFOWQ | 5000_STORE_MESSAGE | 30 | | | | 14 | | | | 4 | | | | 3468000 | | | |
| TPFOWQ | 5000_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| TPFOWQ | 5010_STORE_PRINT_MESSAGE | 37 | | | | 4 | | | | 2 | | | | 362637 | | | |
| TPFOWQ | 5010_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| TPFOWQ | 6000_FINISH_MESSAGE | 18 | | | | 6 | | | | 2 | | | | 106722 | | | |
| TPFOWQ | 6000_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| TPFOWQ | 6010_FINISH_PRINT_MESSAGE | 35 | | | | 11 | | | | 4 | | | | 15571115 | | | |
| TPFOWQ | 6010_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| TPFOWQ | 9000_ERR_00 | 3 | | | | 2 | | | | 2 | | | | 108 | | | |
| TPFOWQ | 9000_ERR_01 | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| TPFOWQ | 9000_ERR_02 | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| TPFOWQ | 9000_ERR_03 | 2 | | | | 1 | | | | 1 | | | | 32 | | | |
| TPFOWQ | 9000_ERR_04 | 2 | | | | 1 | | | | 1 | | | | 32 | | | |
| TPFOWQ | 9000_ERR_05 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_06 | 2 | | | | 1 | | | | 1 | | | | 32 | | | |
| TPFOWQ | 9000_ERR_07 | 2 | | | | 1 | | | | 1 | | | | 32 | | | |
| TPFOWQ | 9000_ERR_08 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_09 | 2 | | | | 1 | | | | 1 | | | | 288 | | | |
| TPFOWQ | 9000_ERR_11 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_12 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_13 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_14 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_15 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_16 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_17 | 2 | | | | 1 | | | | 1 | | | | 72 | | | |
| TPFOWQ | 9000_ERR_18 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_19 | 3 | | | | 2 | | | | 2 | | | | 27 | | | |
| TPFOWQ | 9000_ERR_20 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_21 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_22 | 6 | | | | 1 | | | | 1 | | | | 2904 | | | |
| TPFOWQ | 9000_ERR_30 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_31 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_32 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_33 | 2 | | | | 1 | | | | 1 | | | | 32 | | | |
| TPFOWQ | 9000_ERR_34 | 2 | | | | 1 | | | | 1 | | | | 32 | | | |
| TPFOWQ | 9000_ERR_35 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_36 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_40 | 2 | | | | 1 | | | | 1 | | | | 288 | | | |
| TPFOWQ | 9000_ERR_41 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_42 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |

# TRAIS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| TPFOWQ | 9000_ERR_43 | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| TPFOWQ | 9000_ERR_44 | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| TPFOWQ | 9000_ERR_99 | 6 | | | | 2 | | | | 2 | | | | 4704 | | | |
| TPFOWQ | 9100_ERR_CLEANUP | 17 | | | | 6 | | | | 3 | | | | 6284288 | | | |
| TPFOWQ | 9100_EXIT | 0 | | | | 0 | | | | 0 | | | | 0 | | | |
| NXIEWQ | 000_IE_DRIVER | | 7 | | | | 3 | | | | 2 | | | | 448 | | |
| NXIEWQ | 000_STOP_RUN | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXIEWQ | 100_HSKP | | 38 | | | | 6 | | | | 2 | | | | 30711638 | | |
| NXIEWQ | 100_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXIEWQ | 110_SELECT_RECORDS | | 38 | | | | 19 | | | | 2 | | | | 5429592 | | |
| NXIEWQ | 110_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXIEWQ | 200_REPORT_BUILDER | | 5 | | | | 4 | | | | 2 | | | | 1125 | | |
| NXIEWQ | 200_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXIEWQ | 210_NAF_LOOP | | 5 | | | | 5 | | | | 2 | | | | 2205 | | |
| NXIEWQ | 210_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXIEWQ | 211_APOE_LOOP | | 5 | | | | 6 | | | | 2 | | | | 3645 | | |
| NXIEWQ | 211_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXIEWQ | 212_APOE_TOTAL | | 11 | | | | 2 | | | | 2 | | | | 14256 | | |
| NXIEWQ | 212_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXIEWQ | 2111_APOD_LOOP | | 26 | | | | 10 | | | | 2 | | | | 2402816 | | |
| NXIEWQ | 2111_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXIEWQ | 2112_APOD_TOTAL | | 20 | | | | 4 | | | | 2 | | | | 288000 | | |
| NXIEWQ | 2112_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXIEWQ | 220_NAF_TOTAL | | 13 | | | | 4 | | | | 2 | | | | 15925 | | |
| NXIEWQ | 220_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXIEWQ | 300_WRAPUP | | 7 | | | | 2 | | | | 2 | | | | 9072 | | |
| NXIEWQ | 300_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXIEWQ | 700_HEADERS | | 10 | | | | 7 | | | | 2 | | | | 12960 | | |
| NXIEWQ | 700_HEADERS_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXIEWQ | 700_TRAILER | | 5 | | | | 3 | | | | 2 | | | | 720 | | |
| NXIEWQ | 700_TRAILER_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXIEWQ | 700_NEW_PAGE | | 7 | | | | 4 | | | | 2 | | | | 5488 | | |
| NXIEWQ | 700_NEW_PAGE_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXIEWQ | 700_COMPUTE | | 23 | | | | 2 | | | | 2 | | | | 7394247 | | |
| NXIEWQ | 700_COMPUTE_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD0WQ | 000_DRIVER | | 3 | | | | 2 | | | | 1 | | | | 27 | | |
| NXD0WQ | 000_STOP_RUN | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD0WQ | 100_HSKP | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD0WQ | 100_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD0WQ | 200_READ | | 5 | | | | 3 | | | | 1 | | | | 1125 | | |
| NXD0WQ | 200_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD0WQ | 300_WRAPUP | | 3 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD0WQ | 300_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD1WQ | 000_DRIVER | | 4 | | | | 2 | | | | 2 | | | | 16 | | |
| NXD1WQ | 000_STOP_RUN | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD1WQ | 100_ONHD | | 5 | | | | 2 | | | | 1 | | | | 180 | | |
| NXD1WQ | 100_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD1WQ | 110_READ | | 5 | | | | 3 | | | | 1 | | | | 1125 | | |
| NXD1WQ | 110_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD1WQ | 200_MVMT | | 5 | | | | 2 | | | | 1 | | | | 180 | | |
| NXD1WQ | 200_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD1WQ | 210_READ | | 5 | | | | 3 | | | | 1 | | | | 1125 | | |
| NXD1WQ | 210_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD2WQ | 000_DRIVER | | 7 | | | | 3 | | | | 2 | | | | 252 | | |
| NXD2WQ | 000_STOP_RUN | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD2WQ | 100_TC_ONHAND | | 5 | | | | 2 | | | | 1 | | | | 720 | | |
| NXD2WQ | 100_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD2WQ | 110_READ_WRITE_LOOP | | 10 | | | | 4 | | | | 2 | | | | 12960 | | |
| NXD2WQ | 110_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD2WQ | 200_TC_MOVEMENT | | 5 | | | | 2 | | | | 1 | | | | 720 | | |
| NXD2WQ | 200_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD2WQ | 210_READ_WRITE_LOOP | | 10 | | | | 4 | | | | 2 | | | | 12960 | | |
| NXD2WQ | 210_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD2WQ | 300_COMT | | 5 | | | | 2 | | | | 1 | | | | 720 | | |
| NXD2WQ | 300_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD2WQ | 310_COMT_LOOP | | 10 | | | | 4 | | | | 2 | | | | 12960 | | |

# TRAIS Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| NXD2WQ | 310_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD3WQ | 000_DRIVER | | 3 | | | | 2 | | | | 2 | | | | 12 | | |
| NXD3WQ | 000_STOP_RUN | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD3WQ | 100_TC_COMMERCIAL | | 5 | | | | 2 | | | | 1 | | | | 180 | | |
| NXD3WQ | 100_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD3WQ | 110_READ_WRITE_LOOP | | 6 | | | | 3 | | | | 1 | | | | 1350 | | |
| NXD3WQ | 110_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD3WQ | 200_TC_MOVEMENT | | 5 | | | | 2 | | | | 1 | | | | 180 | | |
| NXD3WQ | 200_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD3WQ | 210_READ_WRITE_LOOP | | 6 | | | | 3 | | | | 1 | | | | 1350 | | |
| NXD3WQ | 210_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD4WQ | 000_DRIVER | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD4WQ | 000_STOP_RUN | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD4WQ | 100_TC_MOVEMENT | | 5 | | | | 2 | | | | 1 | | | | 180 | | |
| NXD4WQ | 100_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD4WQ | 110_READ_WRITE_LOOP | | 6 | | | | 3 | | | | 1 | | | | 1350 | | |
| NXD4WQ | 110_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD5WQ | 000_DRIVER | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD5WQ | 000_STOP_RUN | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD5WQ | 100_SUM | | 5 | | | | 2 | | | | 1 | | | | 180 | | |
| NXD5WQ | 100_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD5WQ | 110_READ_WRITE_LOOP | | 6 | | | | 3 | | | | 1 | | | | 1350 | | |
| NXD5WQ | 110_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD7WQ | 000_DRIVER | | 3 | | | | 2 | | | | 1 | | | | 27 | | |
| NXD7WQ | 000_STOP_RUN | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD7WQ | 100_HSKP | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD7WQ | 100_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD7WQ | 200_READ | | 5 | | | | 3 | | | | 1 | | | | 1125 | | |
| NXD7WQ | 200_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD7WQ | 300_WRAPUP | | 3 | | | | 1 | | | | 1 | | | | 0 | | |
| NXD7WQ | 300_EXIT | | 1 | | | | 1 | | | | 1 | | | | 0 | | |
| NXIFWQ | 000_IF_DRIVER | | | 6 | | | | 3 | | | | 2 | | | | 384 | |
| NXIFWQ | 000_STOP_RUN | | | 1 | | | | 1 | | | | 1 | | | | 0 | |
| NXIFWQ | 100_HOUSEKEEPING | | | 32 | | | | 7 | | | | 2 | | | | 15235200 | |
| NXIFWQ | 100_EXIT | | | 1 | | | | 1 | | | | 1 | | | | 0 | |
| NXIFWQ | 110_SELECT_HISTORY_RECORDS | | | 16 | | | | 11 | | | | 2 | | | | 69696 | |
| NXIFWQ | 110_EXIT | | | 1 | | | | 1 | | | | 1 | | | | 0 | |
| NXIFWQ | 200_REPORT_BUILDER | | | 7 | | | | 6 | | | | 2 | | | | 12348 | |
| NXIFWQ | 200_EXIT | | | 1 | | | | 1 | | | | 1 | | | | 0 | |
| NXIFWQ | 210_CALC_DTL_WEIGHTS_SHIPMENTS | | | 3 | | | | 1 | | | | 1 | | | | 108 | |
| NXIFWQ | 210_EXIT | | | 1 | | | | 1 | | | | 1 | | | | 0 | |
| NXIFWQ | 220_DETAIL_PROCESS | | | 11 | | | | 1 | | | | 1 | | | | 65219 | |
| NXIFWQ | 220_EXIT | | | 1 | | | | 1 | | | | 1 | | | | 0 | |
| NXIFWQ | 230_TOTAL_PROCESS | | | 9 | | | | 2 | | | | 1 | | | | 14400 | |
| NXIFWQ | 230_EXIT | | | 1 | | | | 1 | | | | 1 | | | | 0 | |
| NXIFWQ | 300_WRAPUP | | | 10 | | | | 2 | | | | 2 | | | | 49000 | |
| NXIFWQ | 300_EXIT | | | 1 | | | | 1 | | | | 1 | | | | 0 | |
| NXIFWQ | 700_NEW_PAGE | | | 7 | | | | 4 | | | | 2 | | | | 2800 | |
| NXIFWQ | 700_NEW_PAGE_EXIT | | | 1 | | | | 1 | | | | 1 | | | | 0 | |
| NXIFWQ | 700_HEADERS | | | 8 | | | | 5 | | | | 2 | | | | 6272 | |
| NXIFWQ | 700_HEADERS_EXIT | | | 1 | | | | 1 | | | | 1 | | | | 0 | |
| NXIFWQ | 700_TRAILER | | | 5 | | | | 3 | | | | 2 | | | | 1125 | |
| NXIFWQ | 700_TRAILER_EXIT | | | 1 | | | | 1 | | | | 1 | | | | 0 | |
| NXIFWQ | 700_READ_SORTED_FILE_GFRC | | | 2 | | | | 2 | | | | 1 | | | | 18 | |
| NXIFWQ | 700_READ_SORTED_FILE_GFRC_EXIT | | | 1 | | | | 1 | | | | 1 | | | | 0 | |

UPDATE Subsystem Raw Product Measures

| | | Num of Stmts | | | | McCabe's CC | | | | Max Nesting | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| YG15WQ | 1_CONTROL | 16 | 16 | | | 6 | 6 | | | 2 | 2 | | | 135424 | 135424 | | |
| YG15WQ | 1B_CONTROL_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YG15WQ | 1A_STARS_CALL | 41 | 41 | | | 13 | 13 | | | 2 | 2 | | | 9446400 | 8865225 | | |
| YG15WQ | 1A_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YG15WQ | 2_SET_STARS_ARGUMENTS | 15 | 15 | | | 4 | 4 | | | 2 | 2 | | | 181500 | 181500 | | |
| YG15WQ | 2A_LOAD_FILCODE | 39 | 39 | | | 14 | 14 | | | 2 | 2 | | | 89856 | 89856 | | |
| YG15WQ | 2A_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YG15WQ | 2A1_LOAD_TEST_FILCODE | 39 | 39 | | | 14 | 14 | | | 2 | 2 | | | 78975 | 78975 | | |
| YG15WQ | 2A1_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YG15WQ | 2B_LOAD_OPIDENT | 24 | 24 | | | 8 | 8 | | | 2 | 2 | | | 46464 | 46464 | | |
| YG15WQ | 2B_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YG15WQ | 2B1_LOAD_FILE_RELATED_OPIDENT | 5 | 5 | | | 2 | 2 | | | 2 | 2 | | | 320 | 320 | | |
| YG15WQ | 2B1_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YG15WQ | 2C_LOAD_ADSW | 53 | 53 | | | 16 | 16 | | | 3 | 3 | | | 347733 | 347733 | | |
| YG15WQ | 2C_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YG15WQ | 2D_LOAD_LOCKE | 18 | 18 | | | 7 | 7 | | | 3 | 3 | | | 48672 | 48672 | | |
| YG15WQ | 2D_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YG15WQ | 2E_LOAD_ALLL | 48 | 48 | | | 16 | 16 | | | 2 | 2 | | | 470448 | 470448 | | |
| YG15WQ | 2E_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YG15WQ | 2F_LOAD_KEYNAME | 33 | 33 | | | 12 | 12 | | | 2 | 2 | | | 66825 | 66825 | | |
| YG15WQ | 2F_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YG15WQ | 2G_LOAD_FILCODE | 39 | 39 | | | 14 | 14 | | | 2 | 2 | | | 78975 | 78975 | | |
| YG15WQ | 2G_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YG15WQ | 2G1_LOAD_TEST_FILCODE | 39 | 39 | | | 14 | 14 | | | 2 | 2 | | | 78975 | 78975 | | |
| YG15WQ | 2G1_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YG15WQ | 3_CHECK_REZULTS | 87 | 87 | | | 51 | 51 | | | 3 | 3 | | | 43118592 | 43118592 | | |
| YG15WQ | 3_CHECK_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YG15WQ | 4_DISPLAY_TROUBLE | 14 | 14 | | | 3 | 3 | | | 3 | 3 | | | 4536 | 4536 | | |
| YG15WQ | 4A_DISPLAY_ARGS | 9 | 9 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YG15WQ | 4A_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGABW | 000_DRIVER | 32 | 33 | | | 8 | 10 | | | 2 | 2 | | | 5120000 | 8449188 | | |
| YGABW | 000_CLOSE_FILES | 3 | 3 | | | 2 | 2 | | | 2 | 2 | | | 12 | 12 | | |
| YGABW | 000_DRIVER_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGABW | 100_ABORT_MANIFEST | 56 | 55 | | | 10 | 10 | | | 2 | 2 | | | 99058400 | 91809520 | | |
| YGABW | 100_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGABW | 200_READ_WORK_FILE | 19 | 19 | | | 11 | 11 | | | 3 | 3 | | | 182476 | 182476 | | |
| YGABW | 200_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGABW | 300_ABORT_DETAIL | 29 | 30 | | | 5 | 6 | | | 2 | 2 | | | 4827456 | 6768750 | | |
| YGABW | 300_ABORT_DTL_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGABW | 400_ABORT_PALLET | 22 | 22 | | | 3 | 3 | | | 2 | 2 | | | 2914912 | 2914912 | | |
| YGABW | 400_ABORT_PLT_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGABW | 500_ADD_NEW_HOST_RECORD | 12 | 12 | | | 3 | 3 | | | 3 | 3 | | | 23232 | 23232 | | |
| YGABW | 500_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGABW | 510_REPLACE_OLD_HOST | 16 | 16 | | | 3 | 3 | | | 3 | 3 | | | 156816 | 156816 | | |
| YGABW | 510_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGABW | 600_READ_WRITE | 16 | 16 | | | 7 | 7 | | | 2 | 2 | | | 25600 | 25600 | | |
| YGABW | 600_READ_WRITE_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGABW | 700_TZZ | 16 | 16 | | | 9 | 9 | | | 3 | 3 | | | 1296 | 1296 | | |
| YGABW | 700_TZZ_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGABW | 800_RZLT_CODE_CHECK | 8 | 8 | | | 2 | 2 | | | 2 | 2 | | | 2592 | 2592 | | |
| YGABW | 800_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGABW | 900_SEND_MSG_TO_CONSOLE | 4 | 4 | | | 1 | 1 | | | 1 | 1 | | | 16 | 16 | | |
| YGABW | 900_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGABW | 1000_SEND_MSG | 11 | 11 | | | 4 | 4 | | | 3 | 3 | | | 4851 | 4851 | | |
| YGABW | 1000_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGABW | 1100_BUILD_STATUS_RECORD | 17 | 10 | | | 2 | 1 | | | 2 | 1 | | | 1332800 | 121000 | | |
| YGABW | 1100_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGABW | 1200_DUPE_CONTROL | 11 | 11 | | | 4 | 4 | | | 4 | 4 | | | 27500 | 27500 | | |
| YGABW | 1200_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGABW | 1300_NSN_PROCESS | 8 | 8 | | | 1 | 1 | | | 1 | 1 | | | 3528 | 3528 | | |
| YGABW | 1300_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGABW | 1400_ADD_DETAIL | 7 | 7 | | | 1 | 1 | | | 1 | 1 | | | 5488 | 5488 | | |
| YGABW | 1400_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGABW | 1500_GET_SYS_ID_RECORD | 19 | 19 | | | 4 | 4 | | | 3 | 3 | | | 57475 | 57475 | | |

## UPDATE Subsystem Raw Product Measures

| | | Num of Stmts | | | | McCabe's CC | | | | Max Nesting | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| YGABW | 1500_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGALWQ | 000_DRIVER | 69 | 69 | | 69 | 21 | 21 | | 21 | 3 | 3 | | 3 | 171163125 | 178854900 | | 178854900 |
| YGALWQ | 000_CONTINUE | 5 | 21 | | 21 | 2 | 4 | | 4 | 2 | 3 | | 3 | 1620 | 979776 | | 979776 |
| YGALWQ | 000_CLOSE_FILES | 10 | 10 | | 10 | 6 | 6 | | 6 | 3 | 3 | | 3 | 25000 | 25000 | | 25000 |
| YGALWQ | 000_DRIVER_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 100_LIFT_U | 73 | 78 | | 78 | 16 | 16 | | 16 | 2 | 2 | | 2 | 177652800 | 242437182 | | 242437182 |
| YGALWQ | 100_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 200_MAIN_U | 12 | 12 | | 12 | 10 | 10 | | 10 | 2 | 2 | | 2 | 13068 | 13068 | | 13068 |
| YGALWQ | 200_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 300_BUILD_405 | 13 | 13 | | 13 | 3 | 3 | | 3 | 2 | 2 | | 2 | 67392 | 67392 | | 67392 |
| YGALWQ | 300_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 400_LIFT_DETAIL | 28 | 30 | | 30 | 3 | 3 | | 3 | 2 | 2 | | 2 | 5976432 | 7620480 | | 7620480 |
| YGALWQ | 400_LIFT_DTL_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 410_TCN_CHECK | 25 | 25 | | 25 | 8 | 8 | | 8 | 2 | 2 | | 2 | 202500 | 202500 | | 202500 |
| YGALWQ | 410_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 500_LIFT_PALLET | 18 | 18 | | 18 | 3 | 3 | | 3 | 2 | 2 | | 2 | 793800 | 793800 | | 793800 |
| YGALWQ | 500_LIFT_PLT_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 600_INITIALIZE_RCOMC | 12 | 12 | | 12 | 1 | 1 | | 1 | 1 | 1 | | 1 | 129792 | 129792 | | 129792 |
| YGALWQ | 600_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 700_SEND_405 | 18 | 18 | | 18 | 5 | 5 | | 5 | 3 | 3 | | 3 | 209952 | 209952 | | 209952 |
| YGALWQ | 700_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 800_READ_WRITE | 36 | 36 | | 36 | 12 | 12 | | 12 | 3 | 3 | | 3 | 1192464 | 1192464 | | 1192464 |
| YGALWQ | 800_READ_WRITE_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 810_EDIT_MINI_DIC | 5 | 5 | | 5 | 4 | 4 | | 4 | 2 | 2 | | 2 | 500 | 500 | | 500 |
| YGALWQ | 810_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 820_BALANCE_CHECK_DRIVER | 14 | 14 | | 14 | 7 | 7 | | 7 | 2 | 2 | | 2 | 12600 | 12600 | | 12600 |
| YGALWQ | 820_BALANCE_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 830_PLT_INPUT | 32 | 32 | | 32 | 3 | 3 | | 3 | 2 | 2 | | 2 | 1059968 | 1059968 | | 1059968 |
| YGALWQ | 830_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 840_LOOSE_INPUT | 36 | 36 | | 36 | 4 | 4 | | 4 | 2 | 2 | | 2 | 2663424 | 2663424 | | 2663424 |
| YGALWQ | 840_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 850_TCN_INPUT | 39 | 39 | | 39 | 6 | 6 | | 6 | 3 | 3 | | 3 | 2135484 | 2135484 | | 2135484 |
| YGALWQ | 850_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 860_PLT_BALANCE_CHECK | 30 | 35 | | 35 | 7 | 9 | | 9 | 3 | 3 | | 3 | 192000 | 283500 | | 283500 |
| YGALWQ | 860_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 870_MANFST_BALANCE_CHECK | 28 | 32 | | 32 | 6 | 6 | | 6 | 3 | 3 | | 3 | 548800 | 720000 | | 720000 |
| YGALWQ | 870_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 880_CHECK_TAA_REC | 8 | 8 | | 8 | 5 | 5 | | 5 | 2 | 2 | | 2 | 1152 | 1152 | | 1152 |
| YGALWQ | 880_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 890_TZZ | 14 | 14 | | 14 | 9 | 9 | | 9 | 3 | 3 | | 3 | 1400 | 1400 | | 1400 |
| YGALWQ | 890_TZZ_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 1000_RZLT_CODE_CHECK | 8 | 8 | | 8 | 2 | 2 | | 2 | 2 | 2 | | 2 | 3200 | 4608 | | 4608 |
| YGALWQ | 1000_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 1100_WEIGHT_CONVERT | 14 | 14 | | 14 | 11 | 11 | | 11 | 2 | 2 | | 2 | 1400 | 1400 | | 1400 |
| YGALWQ | 1100_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 1100A_WEIGHT_CONVERT | 13 | 13 | | 13 | 11 | 11 | | 11 | 2 | 2 | | 2 | 468 | 468 | | 468 |
| YGALWQ | 1100A_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 1110_CUBE_CONVERT | 35 | 36 | | 36 | 21 | 22 | | 22 | 2 | 2 | | 2 | 126000 | 186624 | | 186624 |
| YGALWQ | 1110_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 1200_SEND_MSG_TO_CONSOLE | 4 | 4 | | 4 | 1 | 1 | | 1 | 1 | 1 | | 1 | 16 | 16 | | 16 |
| YGALWQ | 1200_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 1400_SEND_MSG | 11 | 11 | | 11 | 4 | 4 | | 4 | 3 | 3 | | 3 | 4851 | 4851 | | 4851 |
| YGALWQ | 1400_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 1500_DELETE_ASIF_MANIFEST | 18 | 39 | | 39 | 2 | 7 | | 7 | 2 | 4 | | 4 | 778752 | 11372400 | | 11372400 |
| YGALWQ | 1500_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 1600_ADD_TRAIS_RECORD | 9 | 11 | | 11 | 3 | 4 | | 4 | 2 | 4 | | 4 | 3969 | 33275 | | 33275 |
| YGALWQ | 1600_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 1700_DUPE_CONTROL | 11 | 3 | | 3 | 4 | 2 | | 2 | 4 | 2 | | 2 | 27500 | 432 | | 432 |
| YGALWQ | 1700_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 1800_SET_LIFT_STATUS | 3 | 18 | | 18 | 2 | 4 | | 4 | 2 | 3 | | 3 | 432 | 54450 | | 54450 |
| YGALWQ | 1800_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGALWQ | 1900_GET_SYS_ID_RECORD | 18 | 10 | | 10 | 4 | 1 | | 1 | 3 | 1 | | 1 | 54450 | 100000 | | 100000 |
| YGALWQ | 1900_EXIT | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 0 | 0 | | 0 |
| YGBRW | 000_CONTROL | 28 | | | | 10 | | | | 3 | | | | 274428 | | | |

## UPDATE Subsystem Raw Product Measures

| | | Num of Stmts | | | | McCabe's CC | | | | Max Nesting | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| YGBRW | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGBRW | 100_BUILD_DETAIL_RECORD | 20 | | | | 2 | | | | 2 | | | | 2857680 | | | |
| YGBRW | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGBRW | 200_BUILD_PALLET_RECORD | 16 | | | | 3 | | | | 2 | | | | 1382976 | | | |
| YGBRW | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 1_START_PROCESSING | 80 | | | | 13 | | | | 2 | | | | 170995520 | | | |
| YGCDW | 1_CONT | 9 | | | | 2 | | | | 2 | | | | 35721 | | | |
| YGCDW | 1_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 2_CALL_UC | 25 | | | | 6 | | | | 2 | | | | 874225 | | | |
| YGCDW | 2_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 3_CALL_15 | 28 | | | | 9 | | | | 4 | | | | 818748 | | | |
| YGCDW | 3_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 3A_RZLT_CODE_CHECK | 9 | | | | 2 | | | | 2 | | | | 6561 | | | |
| YGCDW | 3A_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 4_U_DETAILS | 26 | | | | 7 | | | | 5 | | | | 585000 | | | |
| YGCDW | 4_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 5_ADD_DB_RECORD | 42 | | | | 6 | | | | 3 | | | | 27284712 | | | |
| YGCDW | 5_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 5A_SEARCH_MINI_STATUS | 5 | | | | 3 | | | | 3 | | | | 500 | | | |
| YGCDW | 5A_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 6_DELETE_DB_RECORD | 13 | | | | 3 | | | | 2 | | | | 105300 | | | |
| YGCDW | 6_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 6A_SEARCH_HOST_STATUS | 5 | | | | 3 | | | | 3 | | | | 500 | | | |
| YGCDW | 6A_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 7_COMPARE_RECORD | 82 | | | | 15 | | | | 3 | | | | 152560672 | | | |
| YGCDW | 7_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 8_DISPLAY_COUNTERS | 52 | | | | 6 | | | | 1 | | | | 1684800 | | | |
| YGCDW | 8_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 8A_MOVE | 1 | | | | 1 | | | | 1 | | | | 9 | | | |
| YGCDW | 8A_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 8B_MOVE | 5 | | | | 2 | | | | 1 | | | | 2000 | | | |
| YGCDW | 8B_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 8B1_MOVE | 1 | | | | 1 | | | | 1 | | | | 49 | | | |
| YGCDW | 8B1_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 9_SKIP_COMPARE | 9 | | | | 1 | | | | 1 | | | | 18225 | | | |
| YGCDW | 9_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 10_GET_SYSTEM_ID_REC | 8 | | | | 2 | | | | 2 | | | | 2048 | | | |
| YGCDW | 10_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 15_CHECK_TRANS | 13 | | | | 4 | | | | 4 | | | | 2925 | | | |
| YGCDW | 15_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 16_CHECK_TIME_PASSED | 19 | | | | 5 | | | | 4 | | | | 24624 | | | |
| YGCDW | 16_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 22_DISPLAY_ARGS | 9 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 22_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 23_DISPLAY_RECS | 4 | | | | 2 | | | | 2 | | | | 400 | | | |
| YGCDW | 23_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCDW | 24_ADD_TO_MOVEMENT | 25 | | | | 3 | | | | 2 | | | | 2924100 | | | |
| YGCDW | 24_CONTINUE | 8 | | | | 3 | | | | 2 | | | | 3528 | | | |
| YGCDW | 24_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCPW | 1_START_PROCESSING | 31 | | | | 9 | | | | 2 | | | | 4715100 | | | |
| YGCPW | 1_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCPW | 2_CALL_UC | 25 | | | | 7 | | | | 2 | | | | 874225 | | | |
| YGCPW | 2_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCPW | 3_CALL_15 | 7 | | | | 3 | | | | 2 | | | | 1372 | | | |
| YGCPW | 3A_CALL_15_AGAIN | 19 | | | | 7 | | | | 4 | | | | 205504 | | | |
| YGCPW | 3_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCPW | 3A_RZLT_CODE_CHECK | 13 | | | | 2 | | | | 2 | | | | 8788 | | | |
| YGCPW | 4_U_PALLETS | 27 | | | | 8 | | | | 5 | | | | 995328 | | | |
| YGCPW | 4_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCPW | 5_ADD_DB_RECORD | 29 | | | | 4 | | | | 2 | | | | 10755549 | | | |
| YGCPW | 5_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCPW | 5A_SEARCH_MINI_STATUS | 5 | | | | 3 | | | | 3 | | | | 500 | | | |
| YGCPW | 6_DELETE_DB_RECORD | 12 | | | | 3 | | | | 2 | | | | 76800 | | | |
| YGCPW | 6_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

# UPDATE Subsystem Raw Product Measures

| Program Name | Paragraph (Procedure) Name | Num of Stmts | | | | McCabe's CC | | | | Max Nesting | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| YGCPW | 6A_SEARCH_HOST_STATUS | 5 | | | | 3 | | | | 3 | | | | 500 | | | |
| YGCPW | 7_COMPARE_RECORD | 42 | | | | 8 | | | | 2 | | | | 13502538 | | | |
| YGCPW | 8_DISPLAY_COUNTERS | 46 | | | | 6 | | | | 1 | | | | 1035000 | | | |
| YGCPW | 8A_MOVE | 1 | | | | 1 | | | | 1 | | | | 9 | | | |
| YGCPW | 8B_MOVE | 5 | | | | 2 | | | | 1 | | | | 2000 | | | |
| YGCPW | 8B1_MOVE | 1 | | | | 1 | | | | 1 | | | | 49 | | | |
| YGCPW | 9_SKIP_COMPARE | 9 | | | | 1 | | | | 1 | | | | 18225 | | | |
| YGCPW | 15_CHECK_TRANS | 13 | | | | 4 | | | | 4 | | | | 2925 | | | |
| YGCPW | 15_CHECK_TRANS_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCPW | 16_CHECK_TIME_PASSED | 19 | | | | 5 | | | | 4 | | | | 24624 | | | |
| YGCPW | 16_CHECK_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCPW | 17_DISPLAY_RECS | 4 | | | | 2 | | | | 2 | | | | 144 | | | |
| YGCPW | 17_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCPW | 18_GET_SYSTEM_ID_REC | 10 | | | | 2 | | | | 2 | | | | 10240 | | | |
| YGCPW | 18_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCTW | 1_BEGIN | 22 | | | | 4 | | | | 4 | | | | 71478 | | | |
| YGCTW | 1_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCTW | 2_ADD | 73 | | | | 21 | | | | 3 | | | | 9513433 | | | |
| YGCTW | 2_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCTW | 3_PRINT | 9 | | | | 1 | | | | 1 | | | | 1296 | | | |
| YGCTW | 3_PRINT_TRANS_CNT | 43 | | | | 2 | | | | 2 | | | | 2107 | | | |
| YGCTW | 3_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCTW | 5_SEND_MESSAGE | 28 | | | | 5 | | | | 3 | | | | 444528 | | | |
| YGCTW | 5_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCTW | 6_CHANGE_SYSTEM_STATUS | 18 | | | | 6 | | | | 5 | | | | 71442 | | | |
| YGCTW | 6_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCTW | 7_SET_SYSTEM_STATUS | 15 | | | | 3 | | | | 3 | | | | 18375 | | | |
| YGCTW | 7_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCTW | 8_CALL_15 | 6 | | | | 2 | | | | 2 | | | | 1176 | | | |
| YGCTW | 8_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCTW | 9_RZLT_CHECK | 8 | | | | 2 | | | | 2 | | | | 800 | | | |
| YGCTW | 9_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCTW | 10_READ_SYS_ID_FILE | 15 | | | | 5 | | | | 3 | | | | 73500 | | | |
| YGCTW | 10_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCTW | 11_FIND_PORT | 11 | | | | 3 | | | | 3 | | | | 2816 | | | |
| YGCTW | 11_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCTW | 12_MOVE_DATA | 23 | | | | 2 | | | | 2 | | | | 298908 | | | |
| YGCTW | 12_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCTW | 14_SET_CHANGE_DATE | 4 | | | | 1 | | | | 1 | | | | 4096 | | | |
| YGCTW | 14_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCTW | 00000_BEGIN | | | 28 | 39 | | | 5 | 6 | | | 3 | 5 | | | 252700 | 351975 |
| YGCTW | 00000_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 10000_END_OF_DAY_PROCESS | | | 14 | 14 | | | 2 | 2 | | | 1 | 2 | | | 89600 | 89600 |
| YGCTW | 10000_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 20000_READ_SYS_ID_FILE | | | 16 | 16 | | | 5 | 5 | | | 2 | 3 | | | 129600 | 129600 |
| YGCTW | 20000_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 21000_LOAD_TRANS_TABLE | | | 5 | 5 | | | 2 | 2 | | | 0 | 1 | | | 6480 | 6480 |
| YGCTW | 21000_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 21100_LOAD_SYSID_DATA | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 9 | 9 |
| YGCTW | 21100_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 30000_FIND_PORT | | | 15 | 15 | | | 3 | 3 | | | 2 | 3 | | | 10935 | 10935 |
| YGCTW | 30000_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 40000_ADD | | | 76 | 68 | | | 22 | 18 | | | 2 | 2 | | | 19305216 | 7953552 |
| YGCTW | 40000_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 41000_SEND_MESSAGE | | | 28 | 28 | | | 5 | 5 | | | 2 | 3 | | | 444528 | 444528 |
| YGCTW | 41000_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 42000_CHANGE_SYSTEM_STATUS | | | 19 | 19 | | | 6 | 6 | | | 4 | 5 | | | 75411 | 75411 |
| YGCTW | 42000_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 42100_SET_SYSTEM_STATUS | | | 15 | 15 | | | 3 | 3 | | | 2 | 3 | | | 18375 | 18375 |
| YGCTW | 42100_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 42110_SET_CHANGE_DATE | | | 4 | 4 | | | 1 | 1 | | | 0 | 1 | | | 4096 | 4096 |
| YGCTW | 42110_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 42120_CALL_15 | | | 5 | 5 | | | 2 | 2 | | | 1 | 2 | | | 980 | 980 |
| YGCTW | 42120_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |

UPDATE Subsystem Raw Product Measures

| | | Num of Stmts | | | | McCabe's CC | | | | Max Nesting | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| YGCTW | 50000_STORE_TO_SYS_ID | | | 8 | 8 | | | 3 | 3 | | | 0 | 1 | | | 7200 | 7200 |
| YGCTW | 50000_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 51000_READ_SYS_ID_FILE_AGAIN | | | 14 | 14 | | | 8 | 8 | | | 2 | 3 | | | 50400 | 50400 |
| YGCTW | 51000_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 51100_SEARCH_TRANS_TABLE | | | 3 | 3 | | | 2 | 2 | | | 1 | 2 | | | 192 | 192 |
| YGCTW | 51100_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 51110_STORE_DATA | | | 8 | 8 | | | 3 | 3 | | | 1 | 2 | | | 7200 | 7200 |
| YGCTW | 51110_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 51111_STORE_TRANS_INFO | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 16 | 16 |
| YGCTW | 51111_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 51200_REPLACE_RECORD | | | 9 | 9 | | | 2 | 2 | | | 1 | 2 | | | 3600 | 3600 |
| YGCTW | 51200_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 60000_PROCESS_TRANS_RPT | | | 10 | 10 | | | 4 | 4 | | | 0 | 1 | | | 39690 | 39690 |
| YGCTW | 60000_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 61000_BUILD_AND_SEND | | | 29 | 29 | | | 7 | 7 | | | 2 | 3 | | | 1841616 | 1841616 |
| YGCTW | 61000_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 61100_SET_HEADERS | | | 4 | 11 | | | 2 | 2 | | | 1 | 2 | | | 400 | 107811 |
| YGCTW | 61100_EXIT | | | 0 | 0 | | | 1 | 0 | | | 0 | 0 | | | 0 | 0 |
| YGCTW | 61200_BUILD_DRIVER | | | 5 | 5 | | | 2 | 2 | | | 0 | 1 | | | 8820 | 8820 |
| YGCTW | 61200_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 61210_ACCESS_2ND_DEMENSION | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 16 | 16 |
| YGCTW | 61210_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 61300_SEND_TRANS_RPT | | | 22 | 22 | | | 3 | 3 | | | 2 | 3 | | | 130438 | 130438 |
| YGCTW | 61300_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 61400_SEARCH_PORT_TABLE | | | 6 | 6 | | | 3 | 3 | | | 2 | 3 | | | 384 | 384 |
| YGCTW | 61400_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 70000_RZLT_CHECK | | | 9 | 9 | | | 2 | 2 | | | 1 | 2 | | | 3600 | 3600 |
| YGCTW | 70000_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 80000_FALL_THRU | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGCTW | 80000_EXIT | | | 1 | 1 | | | 1 | 1 | | | 0 | 1 | | | 0 | 0 |
| YGDAW | 100_BEGIN_PROCESSING | 49 | | | | 13 | | | | 3 | | | | 6853924 | | | |
| YGDAW | 100_EOO | 6 | | | | 3 | | | | 3 | | | | 2646 | | | |
| YGDAW | 100_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGDAW | 999_BAD_RESULT | 8 | | | | 2 | | | | 2 | | | | 2592 | | | |
| YGDAW | 999_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGDCW | 100_BEGIN_PROCESSING | 48 | | | | 13 | | | | 3 | | | | 5947392 | | | |
| YGDCW | 100_EOO | 6 | | | | 3 | | | | 3 | | | | 2646 | | | |
| YGDCW | 100_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGDCW | 999_BAD_RESULT | 7 | | | | 2 | | | | 2 | | | | 1372 | | | |
| YGDCW | 999_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGDDW | 100_BEGIN_PROCESSING | 27 | | | | 6 | | | | 4 | | | | 326700 | | | |
| YGDDW | 100_EOO | 6 | | | | 3 | | | | 3 | | | | 2646 | | | |
| YGDDW | 100_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGDDW | 200_ADD_TO_DK | 26 | | | | 4 | | | | 4 | | | | 3521024 | | | |
| YGDDW | 200_ADD_TO_TRAIS | 10 | | | | 3 | | | | 2 | | | | 4410 | | | |
| YGDDW | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGDDW | 999_BAD_RZLT | 8 | | | | 2 | | | | 2 | | | | 1568 | | | |
| YGDDW | 999_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGDMW | 000_DRIVER | 34 | 61 | | | 4 | 12 | | | 2 | 4 | | | 5440000 | 56217600 | | |
| YGDMW | 000_DRIVER_EXIT | | 1 | | | | 1 | | | | 0 | | | | 0 | | |
| YGDMW | 100_ADD_TRAIS_RECORD | 9 | | | | 3 | | | | 2 | | | | 5184 | | | |
| YGDMW | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGDMW | 110_BAD_RZLT_CODE | 7 | | | | 2 | | | | 2 | | | | 1008 | | | |
| YGDMW | 110_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGDMW | 120_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGDMW | 200_BUILD_STATUS_RECORD | 9 | | | | 2 | | | | 2 | | | | 35721 | | | |
| YGDMW | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGDRW | 1_BEGIN_PROCESSING | 6 | | 6 | | 1 | | 1 | | 1 | | 1 | | 1350 | | 1350 | |
| YGDRW | 2_PROCESS | 45 | | 49 | | 11 | | 11 | | 3 | | 3 | | 12545280 | | 17640000 | |
| YGDRW | 2_BY_PASSED_SPAWN_JOBS | 93 | | 93 | | 16 | | 16 | | 3 | | 3 | | 223432500 | | 223432500 | |
| YGDRW | 2_RESULT_CHECK | 4 | | 4 | | 3 | | 3 | | 2 | | 2 | | 256 | | 256 | |
| YGDRW | 3_WRAP_UP | 4 | | 2 | | 1 | | 1 | | 1 | | 1 | | 324 | | 18 | |
| YGDRW | 3_DISPLAY | 10 | | 10 | | 6 | | 6 | | 3 | | 3 | | 9000 | | 9000 | |
| YGDRW | 1_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |

## UPDATE Subsystem Raw Product Measures

| | | Num of Stmts | | | | McCabe's CC | | | | Max Nesting | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| YGDRW | 4_TIME_STAMP_START | 4 | | 4 | | 2 | | 2 | | 2 | | 2 | | 1156 | | 1156 | |
| YGDRW | 4_TIME_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 5_TIME_STAMP_STOP | 4 | | 4 | | 2 | | 2 | | 2 | | 2 | | 1156 | | 1156 | |
| YGDRW | 5_TIME_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 6_SPAWN_DGDH | 35 | | 35 | | 7 | | 7 | | 4 | | 4 | | 3320240 | | 3320240 | |
| YGDRW | 6_SPAWN_DGDH_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 7_SEND_519 | 10 | | 10 | | 3 | | 3 | | 3 | | 3 | | 12250 | | 12250 | |
| YGDRW | 7_519_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 7_SEND_409 | 12 | | 12 | | 7 | | 7 | | 3 | | 3 | | 7500 | | 7500 | |
| YGDRW | 7_409_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 7_SEND_TIPS_PPAL | 19 | | 19 | | 3 | | 3 | | 2 | | 2 | | 1779084 | | 1779084 | |
| YGDRW | 7_PPAL_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 8_REPLACE_CONTROL_RECORD | 8 | | 8 | | 2 | | 2 | | 2 | | 2 | | 18432 | | 18432 | |
| YGDRW | 8_REPLACE_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 8A_GET_CONTROL_RECORD | 8 | | 8 | | 2 | | 2 | | 2 | | 2 | | 24200 | | 24200 | |
| YGDRW | 8A_GET_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 9_BAD_RZLT_CODE | 7 | | 7 | | 2 | | 2 | | 2 | | 2 | | 2268 | | 2268 | |
| YGDRW | 9_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 11_SAVE_PURGE_CHECK | 13 | | 13 | | 4 | | 4 | | 3 | | 3 | | 56628 | | 56628 | |
| YGDRW | 11_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 15_SEND_MSG_TO_CONSOLE | 6 | | 6 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 15_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 17_SPAWN_TCOM | 16 | | 16 | | 4 | | 4 | | 2 | | 2 | | 129600 | | 129600 | |
| YGDRW | 17_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 18_DISPLAY_TO_CONSOL | 3 | | 3 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 18_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 19_DGDH_NOT_SPAWNED | 6 | | 6 | | 1 | | 1 | | 1 | | 1 | | 864 | | 864 | |
| YGDRW | 19_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 20_DGDH_SPAWNED | 15 | | 14 | | 1 | | 1 | | 1 | | 1 | | 6000 | | 5600 | |
| YGDRW | 20_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 22_DGDH_DISPLAYS | 3 | | 3 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 22_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 23_DO_SAVE | 19 | | 19 | | 2 | | 2 | | 2 | | 2 | | 186219 | | 186219 | |
| YGDRW | 23_TAKE_SAVE | 11 | | 11 | | 1 | | 1 | | 1 | | 1 | | 11264 | | 11264 | |
| YGDRW | 23_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 25_CONTROL_RECORD | 13 | | 13 | | 3 | | 3 | | 3 | | 3 | | 67392 | | 67392 | |
| YGDRW | 25_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGDRW | 26_STARS_EOO | 2 | | 2 | | 1 | | 1 | | 1 | | 1 | | 200 | | 200 | |
| YGDRW | 26_EXIT | 1 | | 1 | | 1 | | 1 | | 1 | | 1 | | 0 | | 0 | |
| YGFBWQ | 1_BEGIN_PROCESSING | 81 | | | | 28 | | | | 4 | | | | 86601636 | | | |
| YGFBWQ | 1_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGFBWQ | 2_SEND_TO_MACA | 28 | | | | 5 | | | | 3 | | | | 1097712 | | | |
| YGFBWQ | 2_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGLFWQ | 000_DRIVER | 51 | 56 | | | 17 | 19 | | | 3 | 3 | | | 86455404 | 118065024 | | |
| YGLFWQ | 000_CONTINUE | 21 | 20 | | | 5 | 4 | | | 3 | 3 | | | 734349 | 699380 | | |
| YGLFWQ | 000_CLOSE_FILES | 10 | 10 | | | 6 | 6 | | | 3 | 3 | | | 16000 | 16000 | | |
| YGLFWQ | 000_DRIVER_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 100_LIFT_U | 68 | 66 | | | 13 | 12 | | | 2 | 2 | | | 157313988 | 137618976 | | |
| YGLFWQ | 100_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 200_MAIN_U | 15 | 15 | | | 9 | 9 | | | 2 | 2 | | | 65340 | 65340 | | |
| YGLFWQ | 200_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 210_TRAILER_RECORDS | 6 | 6 | | | 5 | 5 | | | 3 | 3 | | | 600 | 600 | | |
| YGLFWQ | 210_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 210A_STARS_TRAILERS | 15 | 15 | | | 2 | 2 | | | 2 | 2 | | | 121500 | 121500 | | |
| YGLFWQ | 210A_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 300_BUILD_405 | 13 | 13 | | | 3 | 3 | | | 2 | 2 | | | 67392 | 67392 | | |
| YGLFWQ | 300_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 400_LIFT_DETAIL | 42 | 42 | | | 5 | 5 | | | 3 | 3 | | | 23877672 | 23877672 | | |
| YGLFWQ | 400_LIFT_DTL_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 410_TCN_CHECK | 25 | 25 | | | 8 | 8 | | | 2 | 2 | | | 202500 | 202500 | | |
| YGLFWQ | 410_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 500_LIFT_PALLET | 38 | 38 | | | 6 | 6 | | | 3 | 3 | | | 14093478 | 14093478 | | |
| YGLFWQ | 500_LIFT_PLT_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 600_INITIALIZE_RCOMC | 12 | 12 | | | 1 | 1 | | | 1 | 1 | | | 129792 | 129792 | | |

UPDATE Subsystem Raw Product Measures

| | | Num of Stmts | | | | McCabe's CC | | | | Max Nesting | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| YGLFWQ | 600_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 700_SEND_405 | 18 | 18 | | | 5 | 5 | | | 3 | 3 | | | 209952 | 209952 | | |
| YGLFWQ | 700_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 800_READ_WRITE | 34 | 34 | | | 11 | 11 | | | 3 | 3 | | | 971074 | 971074 | | |
| YGLFWQ | 800_READ_WRITE_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 810_EDIT_MINI_DIC | 5 | 5 | | | 4 | 4 | | | 2 | 2 | | | 500 | 500 | | |
| YGLFWQ | 810_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 820_BALANCE_CHECK_DRIVER | 15 | 15 | | | 7 | 7 | | | 2 | 2 | | | 13500 | 13500 | | |
| YGLFWQ | 820_BALANCE_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 830_PLT_INPUT | 18 | 18 | | | 3 | 3 | | | 2 | 2 | | | 490050 | 490050 | | |
| YGLFWQ | 830_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 840_LOOSE_INPUT | 22 | 22 | | | 4 | 4 | | | 2 | 2 | | | 862488 | 862488 | | |
| YGLFWQ | 840_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 850_TCN_INPUT | 32 | 32 | | | 6 | 6 | | | 3 | 3 | | | 1411200 | 1411200 | | |
| YGLFWQ | 850_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 860_PLT_BALANCE_CHECK | 26 | 33 | | | 6 | 9 | | | 3 | 3 | | | 109850 | 211200 | | |
| YGLFWQ | 860_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 870_MANFST_BALANCE_CHECK | 24 | 30 | | | 5 | 6 | | | 2 | 3 | | | 345600 | 588000 | | |
| YGLFWQ | 870_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 880_CHECK_TAA_REC | 11 | 11 | | | 5 | 5 | | | 2 | 2 | | | 1584 | 1584 | | |
| YGLFWQ | 880_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 890_TZZ | 14 | 14 | | | 9 | 9 | | | 3 | 3 | | | 1400 | 1400 | | |
| YGLFWQ | 890_TZZ_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 1000_RZLT_CODE_CHECK | 8 | 8 | | | 2 | 2 | | | 2 | 2 | | | 5408 | 5408 | | |
| YGLFWQ | 1000_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 1100_WEIGHT_CONVERT | 14 | 14 | | | 11 | 11 | | | 2 | 2 | | | 1400 | 1400 | | |
| YGLFWQ | 1100_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 1100A_WEIGHT_CONVERT | 13 | 13 | | | 11 | 11 | | | 2 | 2 | | | 468 | 468 | | |
| YGLFWQ | 1100A_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 1110_CUBE_CONVERT | 34 | 36 | | | 21 | 22 | | | 2 | 2 | | | 122400 | 186624 | | |
| YGLFWQ | 1110_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 1200_SEND_MSG_TO_CONSOLE | 4 | 4 | | | 1 | 1 | | | 1 | 1 | | | 16 | 16 | | |
| YGLFWQ | 1200_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 1400_SEND_MSG | 11 | 11 | | | 4 | 4 | | | 3 | 3 | | | 4851 | 4851 | | |
| YGLFWQ | 1400_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 1500_BUILD_STATUS_RECORD | 12 | 9 | | | 2 | 1 | | | 2 | 1 | | | 164268 | 72900 | | |
| YGLFWQ | 1500_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 1600_DTL_TAC | 5 | 5 | | | 2 | 2 | | | 2 | 2 | | | 180 | 180 | | |
| YGLFWQ | 1600_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 1700_CALL_YGTFWQ | 8 | 8 | | | 1 | 1 | | | 1 | 1 | | | 31752 | 31752 | | |
| YGLFWQ | 1700_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 1800_DUPE_CONTROL | 11 | 11 | | | 4 | 4 | | | 4 | 4 | | | 27500 | 27500 | | |
| YGLFWQ | 1800_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 1900_SET_LIFT_STATUS | 3 | 3 | | | 2 | 2 | | | 2 | 2 | | | 432 | 432 | | |
| YGLFWQ | 1900_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGLFWQ | 2000_GET_SYS_ID_RECORD | 18 | 18 | | | 4 | 4 | | | 3 | 3 | | | 54450 | 54450 | | |
| YGLFWQ | 2000_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGOBW | 000_DRIVER | 156 | 156 | | | 83 | 83 | | | 2 | 2 | | | 2.752E+09 | 2.752E+09 | | |
| YGOBW | 000_A_SEND_REPORT | 34 | 32 | | | 11 | 11 | | | 2 | 2 | | | 705024 | 663552 | | |
| YGOBW | 000_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGOBW | 020_REPORT_DRIVER | 17 | 17 | | | 8 | 8 | | | 2 | 2 | | | 20825 | 20825 | | |
| YGOBW | 020_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGOBW | 021_PLT_INPUT | 28 | 28 | | | 5 | 5 | | | 3 | 3 | | | 2274300 | 2274300 | | |
| YGOBW | 021_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGOBW | 022_LOOSE_INPUT | 32 | 32 | | | 6 | 6 | | | 3 | 3 | | | 4147200 | 4147200 | | |
| YGOBW | 022_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGOBW | 023_TCN_INPUT | 46 | 46 | | | 8 | 8 | | | 3 | 3 | | | 5994766 | 5994766 | | |
| YGOBW | 023_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGOBW | 030_PLT_BALANCE_CHECK | 31 | 36 | | | 9 | 12 | | | 3 | 3 | | | 642816 | 1040400 | | |
| YGOBW | 030_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGOBW | 040_MANFST_BALANCE_CHECK | 23 | 23 | | | 8 | 8 | | | 3 | 3 | | | 314847 | 314847 | | |
| YGOBW | 040_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGOBW | 050_GET_INPUT | 3 | 3 | | | 2 | 2 | | | 1 | 1 | | | 108 | 108 | | |
| YGOBW | 050_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |

# UPDATE Subsystem Raw Product Measures

| | | Num of Stmts | | | | McCabe's CC | | | | Max Nesting | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | January | April | Baseline | October | January | April | Baseline | October | January | April | Baseline | October | January | April |
| YGOBW | 060_WRITE_DRIVER | 8 | 8 | | | 3 | 3 | | | 2 | 2 | | | 1152 | 1152 | | |
| YGOBW | 060_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGOBW | 070_WRITE_PLT_RECORD | 8 | 8 | | | 5 | 5 | | | 1 | 1 | | | 3200 | 3200 | | |
| YGOBW | 070_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGOBW | 080_WRITE_DETAILS | 9 | 9 | | | 4 | 4 | | | 2 | 2 | | | 7056 | 7056 | | |
| YGOBW | 080_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGOBW | 100_WEIGHT_CONVERT | 27 | 27 | | | 11 | 14 | | | 2 | 2 | | | 314928 | 314928 | | |
| YGOBW | 100_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGOBW | 100A_WEIGHT_CONVERT | 23 | 23 | | | 11 | 12 | | | 2 | 2 | | | 100188 | 100188 | | |
| YGOBW | 100A_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGOBW | 110_CUBE_CONVERT | 35 | 35 | | | 21 | 22 | | | 2 | 2 | | | 343035 | 343035 | | |
| YGOBW | 110_CONVERT_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGOBW | 110_EDIT_TAA | 16 | 16 | | | 7 | 7 | | | 2 | 2 | | | 9216 | 9216 | | |
| YGOBW | 110_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGOBW | 120_MILSTAMP_ERROR | 9 | 9 | | | 2 | 2 | | | 2 | 2 | | | 8100 | 8100 | | |
| YGOBW | 120_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGOBW | 130_GET_SYS_ID_RECORD | 18 | 18 | | | 4 | 4 | | | 3 | 3 | | | 54450 | 54450 | | |
| YGOBW | 130_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGOBW | 131_CHECK_4_PALLETIZED | 9 | 9 | | | 3 | 3 | | | 2 | 2 | | | 20736 | 20736 | | |
| YGOBW | 131_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 000_L2_DRIVER | 23 | 23 | | | 9 | 9 | | | 2 | 2 | | | 100188 | 100188 | | |
| YGORW | 000_EXIT_PROGRAM | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 100_SLEW_TRANSLATOR_HSKP | 13 | 13 | | | 3 | 3 | | | 2 | 2 | | | 1872 | 1872 | | |
| YGORW | 100_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 200_SLEW_TRANSLATOR_DRIVER | 5 | 5 | | | 3 | 3 | | | 2 | 2 | | | 180 | 180 | | |
| YGORW | 200_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 210_READ_REPORT_FILE | 8 | 8 | | | 3 | 3 | | | 2 | 2 | | | 800 | 800 | | |
| YGORW | 210_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 220_TRANSLATE_AND_WRITE | 50 | 50 | | | 11 | 11 | | | 4 | 4 | | | 583200 | 583200 | | |
| YGORW | 220_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 221_SPACE_LINE | 6 | 6 | | | 2 | 2 | | | 1 | 1 | | | 3456 | 3456 | | |
| YGORW | 221_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 222_REPORT_DATA | 5 | 5 | | | 3 | 3 | | | 2 | 2 | | | 2420 | 2420 | | |
| YGORW | 222_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 300_SLEW_TRANSLATOR_WRAPUP | 2 | 2 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 300_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 400_RPT_TRANSMITTER_HSKP | 19 | 19 | | | 1 | 1 | | | 1 | 1 | | | 190000 | 190000 | | |
| YGORW | 400_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 500_RPT_TRANSMITTER_DRIVER | 10 | 10 | | | 8 | 8 | | | 2 | 2 | | | 9000 | 9000 | | |
| YGORW | 500_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 510_MINIS_SEND_DRIVER | 11 | 11 | | | 8 | 8 | | | 2 | 2 | | | 6875 | 6875 | | |
| YGORW | 510_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 511_FIRST_READ | 10 | 10 | | | 2 | 2 | | | 1 | 1 | | | 5760 | 5760 | | |
| YGORW | 511_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 512_LOAD_MINI_PRT_HDR | 16 | 16 | | | 2 | 2 | | | 2 | 2 | | | 36864 | 36864 | | |
| YGORW | 512_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 513_LOAD_REPORT_DATA | 20 | 20 | | | 9 | 9 | | | 2 | 2 | | | 141120 | 141120 | | |
| YGORW | 513_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 514_MESSAGE_CONTROL | 16 | 16 | | | 12 | 12 | | | 2 | 2 | | | 9216 | 9216 | | |
| YGORW | 514_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 515_FORMAT_CALL_RBTCHO | 33 | 32 | | | 7 | 7 | | | 2 | 2 | | | 1427712 | 1384448 | | |
| YGORW | 515_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 520_BD700_SEND_DRIVER | 5 | 3 | | | 3 | 2 | | | 2 | 1 | | | 320 | 48 | | |
| YGORW | 520_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 521_FIRST_READ | 15 | 15 | | | 5 | 5 | | | 2 | 2 | | | 26460 | 26460 | | |
| YGORW | 521_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 522_READ_SEND_RECORD | 23 | 23 | | | 6 | 6 | | | 2 | 2 | | | 139932 | 139932 | | |
| YGORW | 522_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 700_ERROR_DISPLAY | 12 | 12 | | | 1 | 1 | | | 1 | 1 | | | 9408 | 9408 | | |
| YGORW | 700_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 800_DROPPED_THRU | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGORW | 800_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGPAW | 000_CONTROL | 44 | | | | 8 | | | | 2 | | | | 3207600 | | | |
| YGPAW | 100_EOO | 6 | | | | 3 | | | | 3 | | | | 2646 | | | |

## UPDATE Subsystem Raw Product Measures

| | | Num of Stmts | | | | McCabe's CC | | | | Max Nesting | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| YGPAW | 100_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGPAW | 999_BAD_RESULT | 8 | | | | 2 | | | | 2 | | | | 2048 | | | |
| YGPAW | 999_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGPCW | 000_CONTROL | 42 | | | | 8 | | | | 2 | | | | 3061800 | | | |
| YGPCW | 100_EOO | 6 | | | | 3 | | | | 3 | | | | 2646 | | | |
| YGPCW | 100_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGPCW | 999_BAD_RESULT | 7 | | | | 2 | | | | 2 | | | | 1372 | | | |
| YGPCW | 999_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGPDW | 000_CONTROL | 24 | | | | 4 | | | | 3 | | | | 185856 | | | |
| YGPDW | 100_EOO | 6 | | | | 3 | | | | 3 | | | | 2646 | | | |
| YGPDW | 100_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGPDW | 999_BAD_RZLT | 8 | | | | 2 | | | | 2 | | | | 1152 | | | |
| YGPDW | 999_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGPFWQ | 1_START_PROCESSING | 43 | 43 | | | 9 | 9 | | | 2 | 2 | | | 3973888 | 4486147 | | |
| YGPFWQ | 1_EXIT_PROGRAM | 2 | 2 | | | 2 | 2 | | | 2 | 2 | | | 2 | 2 | | |
| YGPFWQ | 1B_EXIT_PROGRAM | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGPFWQ | 2_PURGE_DRIVER | 23 | 23 | | | 8 | 8 | | | 2 | 2 | | | 1123343 | 1123343 | | |
| YGPFWQ | 2_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGPFWQ | 4_DELETE_RECORD | 6 | 17 | | | 2 | 4 | | | 2 | 4 | | | 3456 | 49572 | | |
| YGPFWQ | 4_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGPFWQ | 5_DISPLAY_ARGS | 9 | 9 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGPFWQ | 5_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGPFWQ | 6_RELEASE_REC | 5 | 5 | | | 4 | 4 | | | 2 | 2 | | | 2880 | 2880 | | |
| YGPFWQ | 6_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGPFWQ | 8_STARS_RESULT | 71 | 71 | | | 32 | 32 | | | 2 | 2 | | | 13916 | 18176 | | |
| YGPFWQ | 8_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGPFWQ | 9_CALL_YG15WQ | 2 | 2 | | | 1 | 1 | | | 1 | 1 | | | 648 | 800 | | |
| YGPFWQ | 9_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGPFWQ | 10_RTOS_GONE | 3 | 3 | | | 1 | 1 | | | 1 | 1 | | | 432 | 675 | | |
| YGPFWQ | 10_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGPFWQ | 11_OPS_CONSOLE | 7 | 7 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGPFWQ | 11_EXIT | 1 | 1 | | | 1 | 1 | | | 1 | 1 | | | 0 | 0 | | |
| YGPTWQ | 1_BEGIN_PROCESSING | 27 | | | | 9 | | | | 3 | | | | 270000 | | | |
| YGPTWQ | 1_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGPTWQ | 2_EDIT_ARGS | 28 | | | | 19 | | | | 4 | | | | 12348 | | | |
| YGPTWQ | 2_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGPTWQ | 3_BUILD_TABLE_DRIVER | 4 | | | | 3 | | | | 1 | | | | 1024 | | | |
| YGPTWQ | 3_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGPTWQ | 4_LOAD_TABLE | 17 | | | | 6 | | | | 2 | | | | 287300 | | | |
| YGPTWQ | 4_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGPTWQ | 5_FIND_BOTH | 12 | | | | 3 | | | | 2 | | | | 15552 | | | |
| YGPTWQ | 5_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGPTWQ | 6_SEARCH_T1_TABLE | 5 | | | | 3 | | | | 1 | | | | 1280 | | | |
| YGPTWQ | 6_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGTFWQ | 000_DRIVER | 21 | | | | 3 | | | | 2 | | | | 75600 | | | |
| YGTFWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGTFWQ | 010_READ | 13 | | | | 2 | | | | 1 | | | | 7488 | | | |
| YGTFWQ | 010_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGTFWQ | 020_WRITE | 10 | | | | 2 | | | | 1 | | | | 810 | | | |
| YGTFWQ | 020_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGTFWQ | 030_REWRITE | 9 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGTFWQ | 030_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGUCW | 100_BEGIN_PROCESSING | 19 | | 19 | 19 | 12 | | 12 | 12 | 3 | | 3 | 3 | 68400 | | 68400 | 68400 |
| YGUCW | 300_CALL_STGTBL | 35 | | 35 | 35 | 12 | | 12 | 12 | 3 | | 3 | 3 | 3472875 | | 3472875 | 3472875 |
| YGUCW | 310_OTS_NULL_CALL | 32 | | 32 | 32 | 12 | | 12 | 12 | 3 | | 3 | 3 | 1216800 | | 1216800 | 1216800 |
| YGUCW | 320_I_CALL | 159 | | 163 | 163 | 42 | | 43 | 43 | 5 | | 5 | 5 | 1.887E+09 | | 2.07E+09 | 2.07E+09 |
| YGUCW | 300_EXIT | 2 | | 2 | 2 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 400_INITIALIZE_RTOS_UTILITIES | 57 | | 57 | 57 | 6 | | 6 | 6 | 3 | | 3 | 3 | 19976448 | | 19976448 | 19976448 |
| YGUCW | 400_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 500_SEND_104 | 12 | | 12 | 12 | 5 | | 5 | 5 | 3 | | 3 | 3 | 78732 | | 78732 | 78732 |
| YGUCW | 500_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 510_SEND_TO_MINIS | 3 | | 3 | 3 | 1 | | 1 | 1 | 1 | | 1 | 1 | 108 | | 108 | 108 |
| YGUCW | 510_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 600_DO_SAVE | 21 | | 21 | 21 | 2 | | 2 | 2 | 2 | | 2 | 2 | 170100 | | 170100 | 170100 |

## UPDATE Subsystem Raw Product Measures

| | | Num of Stmts | | | | McCabe's CC | | | | Max Nesting | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| YGUCW | 600_TAKE_SAVE | 11 | | 11 | 11 | 1 | | 1 | 1 | 1 | | 1 | 1 | 11264 | | 11264 | 11264 |
| YGUCW | 600_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 700_SET_UP_TIME | 18 | | 18 | 18 | 4 | | 4 | 4 | 2 | | 2 | 2 | 629442 | | 629442 | 629442 |
| YGUCW | 700_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 710_SET_PORT | 16 | | 16 | 16 | 4 | | 4 | 4 | 2 | | 2 | 2 | 40000 | | 40000 | 40000 |
| YGUCW | 710_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 800_RETURN_TO_CRT | 11 | | 11 | 11 | 2 | | 2 | 2 | 2 | | 2 | 2 | 118976 | | 118976 | 118976 |
| YGUCW | 800_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 999_BAD_RCOMC_CALL | 8 | | 8 | 8 | 2 | | 2 | 2 | 2 | | 2 | 2 | 2592 | | 2592 | 2592 |
| YGUCW | 999_BAD_RCOMC_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 999_BAD_STARS_CALL | 7 | | 7 | 7 | 2 | | 2 | 2 | 2 | | 2 | 2 | 5488 | | 5488 | 5488 |
| YGUCW | 999_STARS_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 1000_BUILD_TABLE | 17 | | 17 | 17 | 8 | | 8 | 8 | 2 | | 2 | 2 | 49572 | | 49572 | 49572 |
| YGUCW | 1000_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 1200_MOVE_TIME | 12 | | 12 | 12 | 5 | | 5 | 5 | 2 | | 2 | 2 | 37632 | | 37632 | 37632 |
| YGUCW | 1200_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 1300_BUILD_STAT_SCREEN | 9 | | 9 | 9 | 2 | | 2 | 2 | 2 | | 2 | 2 | 15876 | | 15876 | 15876 |
| YGUCW | 1300_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 1400_CONTROL_RECORD | 13 | | 15 | 15 | 3 | | 3 | 3 | 3 | | 3 | 3 | 67392 | | 181500 | 181500 |
| YGUCW | 1400_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 1600_STORE_STATS_DRIVER | 7 | | 7 | 7 | 3 | | 3 | 3 | 1 | | 1 | 1 | 5488 | | 5488 | 5488 |
| YGUCW | 1600_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 1700_READ_SYS_ID_FILE | 11 | | 11 | 11 | 5 | | 5 | 5 | 3 | | 3 | 3 | 6336 | | 6336 | 6336 |
| YGUCW | 1700_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 1800_SEARCH_STAT_TABLE | 4 | | 4 | 4 | 3 | | 3 | 3 | 1 | | 1 | 1 | 900 | | 900 | 900 |
| YGUCW | 1800_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 1900_STORE_DATA | 2 | | 2 | 2 | 1 | | 1 | 1 | 1 | | 1 | 1 | 72 | | 72 | 72 |
| YGUCW | 1900_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGUCW | 2000_REPLACE_SYS_ID_RECORD | 7 | | 7 | 7 | 2 | | 2 | 2 | 2 | | 2 | 2 | 2268 | | 2268 | 2268 |
| YGUCW | 2000_EXIT | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 0 |
| YGMPW | 000_BEGIN_PROCESSING | | | | 13 | | | | 3 | | | | 0 | | | | 16848 |
| YGMPW | 100_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGMPW | 200_READ_DP_FILE | | | | 19 | | | | 8 | | | | 1 | | | | 238336 |
| YGMPW | 200_CONTINUE | | | | 8 | | | | 4 | | | | 2 | | | | 1152 |
| YGMPW | 200_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 100_BEGIN_PROCESSING | | | | 28 | | | | 9 | | | | 1 | | | | 716800 |
| YGPRW | 050_LIST_PORTS | | | | 32 | | | | 8 | | | | 1 | | | | 2163200 |
| YGPRW | 050_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 060_PROCESS_LIST | | | | 18 | | | | 8 | | | | 2 | | | | 139392 |
| YGPRW | 060_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 065_WRITE_REPORT | | | | 12 | | | | 3 | | | | 1 | | | | 21168 |
| YGPRW | 065_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 100_READ_NEW_PORT | | | | 53 | | | | 11 | | | | 1 | | | | 25453197 |
| YGPRW | 100_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 100_EXIT_PROGRAM | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 200_GET_MFST_REC | | | | 54 | | | | 13 | | | | 2 | | | | 4024566 |
| YGPRW | 200_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 210_WRITE_DATA | | | | 7 | | | | 3 | | | | 0 | | | | 4375 |
| YGPRW | 210_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 300_SEND_REPORT | | | | 7 | | | | 2 | | | | 0 | | | | 14175 |
| YGPRW | 300_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 400_BOJ_CALL | | | | 8 | | | | 2 | | | | 1 | | | | 9800 |
| YGPRW | 400_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 500_BAD_STARS_CALL | | | | 3 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 500_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 600_EOJ_CALL | | | | 3 | | | | 1 | | | | 0 | | | | 1200 |
| YGPRW | 600_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 700_MOVE_DATA | | | | 25 | | | | 4 | | | | 2 | | | | 245025 |
| YGPRW | 700_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 900_MOVE_INPUT | | | | 22 | | | | 7 | | | | 1 | | | | 17248 |
| YGPRW | 900_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 1000_FIND_PCN | | | | 3 | | | | 2 | | | | 1 | | | | 48 |
| YGPRW | 1000_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 1010_FIND_PCN | | | | 3 | | | | 2 | | | | 1 | | | | 48 |

UPDATE Subsystem Raw Product Measures

| | | Num of Stmts | | | | McCabe's CC | | | | Max Nesting | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April | Baseline | October | Januaray | April |
| YGPRW | 1010_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 1100_DISPLAY_DATA | | | | 20 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 1100_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 1200_READ_PORT_FILE | | | | 2 | | | | 2 | | | | 0 | | | | 18 |
| YGPRW | 1200_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 1400_WRITE_HEADERS | | | | 19 | | | | 3 | | | | 2 | | | | 9196 |
| YGPRW | 1400_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPRW | 1500_WRITE_HEADERS | | | | 9 | | | | 3 | | | | 2 | | | | 5184 |
| YGPRW | 1500_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPMW | 100_BEGIN_PROCESSING | | | | 41 | | | | 8 | | | | 1 | | | | 1511424 |
| YGPMW | 100_EXIT_PROGRAM | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPMW | 200_READ_WRITE | | | | 40 | | | | 12 | | | | 3 | | | | 13829760 |
| YGPMW | 200_CALL_YG15WQ | | | | 27 | | | | 10 | | | | 3 | | | | 657072 |
| YGPMW | 200_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPMW | 400_BOJ_CALL | | | | 8 | | | | 2 | | | | 1 | | | | 9800 |
| YGPMW | 400_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPMW | 500_BAD_STARS_CALL | | | | 4 | | | | 1 | | | | 0 | | | | 0 |
| YGPMW | 500_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPMW | 600_EOJ_CALL | | | | 2 | | | | 1 | | | | 0 | | | | 288 |
| YGPMW | 600_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPMW | 700_BUILD_APO_TABLE | | | | 15 | | | | 4 | | | | 1 | | | | 77760 |
| YGPMW | 700_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPMW | 800_SEARCH_TABLE | | | | 9 | | | | 3 | | | | 1 | | | | 5625 |
| YGPMW | 800_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |
| YGPMW | 900_DISPLAY_REC | | | | 2 | | | | 2 | | | | 1 | | | | 0 |
| YGPMW | 900_EXIT | | | | 1 | | | | 1 | | | | 0 | | | | 0 |

# MACA Subsystem Raw Product measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest Lvl | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April |
| YG12WQ | 100_CONTROL | 17 | | | | 6 | | | | 2 | | | | 74052 | | | |
| YG12WQ | 100_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG12WQ | 200_SET_STARS_ARGUMENTS | 17 | | | | 6 | | | | 3 | | | | 166617 | | | |
| YG12WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG12WQ | 210_LOAD_FILCODE | 26 | | | | 18 | | | | 7 | | | | 5850 | | | |
| YG12WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG12WQ | 220_LOAD_OPIDENT | 7 | | | | 3 | | | | 3 | | | | 1008 | | | |
| YG12WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG12WQ | 230_LOAD_OPERATION | 33 | | | | 13 | | | | 13 | | | | 2673 | | | |
| YG12WQ | 230_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG12WQ | 240_LOAD_LOCK | 11 | | | | 5 | | | | 5 | | | | 891 | | | |
| YG12WQ | 240_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG12WQ | 250_LOAD_ALLL | 33 | | | | 16 | | | | 16 | | | | 2673 | | | |
| YG12WQ | 250_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG12WQ | 260_LOAD_KEY_NAME | 21 | | | | 11 | | | | 10 | | | | 3024 | | | |
| YG12WQ | 260_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG12WQ | 300_STARS_CALL | 23 | | | | 9 | | | | 9 | | | | 745200 | | | |
| YG12WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG12WQ | 400_CHECK_RESULTS | 72 | | | | 51 | | | | 3 | | | | 705672 | | | |
| YG12WQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG12WQ | 500_DISPLAY_TROUBLE | 21 | | | | 5 | | | | 4 | | | | 31941 | | | |
| YG12WQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG12WQ | 600_MOVE_ARGS | 17 | | | | 1 | | | | 1 | | | | 108800 | | | |
| YG12WQ | 600_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG43WQ | 000_START | 1 | | | | 1 | | | | 1 | | | | 16 | | | |
| YG43WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG44WQ | 000_START | 16 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG44WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG44WQ | 100_INIT_HDR8 | 1 | | | | 1 | | | | 1 | | | | 36 | | | |
| YG44WQ | 199_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG44WQ | 200_INIT_HDR10 | 1 | | | | 1 | | | | 1 | | | | 16 | | | |
| YG44WQ | 299_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG44WQ | 300_REPORT | 5 | | | | 1 | | | | 1 | | | | 125 | | | |
| YG44WQ | 310_READ | 36 | | | | 10 | | | | 3 | | | | 4284900 | | | |
| YG44WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG44WQ | 320_CONSIGNOR_CHANGE | 19 | | | | 6 | | | | 2 | | | | 260091 | | | |
| YG44WQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG44WQ | 330_FINISH_HEADER8 | 15 | | | | 5 | | | | 2 | | | | 47040 | | | |
| YG44WQ | 330_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG44WQ | 335_WRITE_HDR8 | 6 | | | | 2 | | | | 1 | | | | 1350 | | | |
| YG44WQ | 335_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG44WQ | 500_WRITE_HEADER | 16 | | | | 9 | | | | 2 | | | | 20736 | | | |
| YG44WQ | 599_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG44WQ | 600_INIT_HEADER | 1 | | | | 1 | | | | 1 | | | | 9 | | | |
| YG44WQ | 699_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG44WQ | 700_SAVE_HEADER | 9 | | | | 1 | | | | 1 | | | | 14400 | | | |
| YG44WQ | 799_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG44WQ | 800_WRITE_SUMMARY | 28 | | | | 27 | | | | 1 | | | | 28672 | | | |
| YG44WQ | 899_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG44WQ | 930_SAVE_HEADER | 13 | | | | 1 | | | | 1 | | | | 29952 | | | |
| YG44WQ | 930_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG44WQ | 1000_DD_HDR10 | 8 | | | | 2 | | | | 1 | | | | 19208 | | | |
| YG44WQ | 1000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG44WQ | 1010_WRITE_DD_HEADER | 7 | | | | 6 | | | | 1 | | | | 3087 | | | |
| YG44WQ | 1010_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YG44WQ | 1020_WRITE_DD_HDR8 | 12 | | | | 2 | | | | 1 | | | | 62208 | | | |
| YG44WQ | 1020_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGADWQ | 000_INITIALIZE | 1 | | | 4 | 1 | | | 2 | 1 | | | 2 | 0 | | | 16 |
| YGADWQ | 010_DRIVER_PARA | 23 | | | 34 | 20 | | | 24 | 5 | | | 5 | 178112 | | | 827424 |
| YGADWQ | 090_EXIT_PROGRAM | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGADWQ | 100_ADD_TRAILER | 7 | | | 7 | 1 | | | 1 | 1 | | | 1 | 2800 | | | 2800 |
| YGADWQ | 105_GET_PRIME | 32 | | | 44 | 5 | | | 8 | 2 | | | 3 | 3276800 | | | 6353600 |

## MACA Subsystem Raw Product measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest Lvl | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April |
| YGADWQ | 110_BUILD_TRAILER | 12 | | | 12 | 2 | | | 2 | 2 | | | 2 | 47628 | | | 47628 |
| YGADWQ | 120_ADD_TRAILER | 16 | | | 16 | 4 | | | 4 | 2 | | | 2 | 36864 | | | 36864 |
| YGADWQ | 130_SEND_TO_MINIS | 3 | | | 3 | 1 | | | 1 | 1 | | | 1 | 48 | | | 48 |
| YGADWQ | 190_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGADWQ | 200_ADD_DETAIL | 13 | | | 21 | 1 | | | 5 | 1 | | | 2 | 206388 | | | 873936 |
| YGADWQ | 210_ADD | 23 | | | 27 | 4 | | | 6 | 2 | | | 3 | 517500 | | | 1190700 |
| YGADWQ | 220_SEND_TO_MINIS | 2 | | | 2 | 1 | | | 1 | 1 | | | 1 | 18 | | | 18 |
| YGADWQ | 290_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGADWQ | 291_GET_DUP | 18 | | | 18 | 4 | | | 4 | 3 | | | 3 | 118098 | | | 118098 |
| YGADWQ | 291_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGADWQ | 293_MOVE_TIME | 9 | | | 9 | 3 | | | 3 | 2 | | | 2 | 20736 | | | 20736 |
| YGADWQ | 293_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGADWQ | 300_CHANGE_DETAIL | 5 | | | 5 | 1 | | | 1 | 1 | | | 1 | 720 | | | 720 |
| YGADWQ | 310_GET_DETAIL | 27 | | | 39 | 4 | | | 7 | 2 | | | 3 | 395307 | | | 797511 |
| YGADWQ | 320_UPDATE_ADVANCE | 16 | | | 16 | 3 | | | 3 | 2 | | | 2 | 313600 | | | 313600 |
| YGADWQ | 330_REPLACE_ADVANCE | 11 | | | 11 | 3 | | | 3 | 2 | | | 2 | 13475 | | | 13475 |
| YGADWQ | 340_SEND_TO_MINIS | 3 | | | 3 | 1 | | | 1 | 1 | | | 1 | 192 | | | 192 |
| YGADWQ | 390_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGADWQ | 400_COUNTER | 10 | | | 10 | 12 | | | 12 | 5 | | | 5 | 3240 | | | 3240 |
| YGADWQ | 400_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGADWQ | 410_COUNTER | 7 | | | 7 | 14 | | | 14 | 4 | | | 4 | 448 | | | 448 |
| YGADWQ | 410_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGADWQ | 500_COUNTER | 10 | | | 10 | 12 | | | 12 | 5 | | | 5 | 3240 | | | 3240 |
| YGADWQ | 500_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGADWQ | 510_COUNTER | 7 | | | 7 | 14 | | | 14 | 4 | | | 4 | 448 | | | 448 |
| YGADWQ | 510_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGADWQ | 600_COMPUTE_ETA | 103 | | | 103 | 35 | | | 35 | 2 | | | 2 | 33372 | | | 33372 |
| YGADWQ | 600_CONTINUE | 14 | | | 14 | 4 | | | 4 | 3 | | | 3 | 5600 | | | 5600 |
| YGADWQ | 600_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGADWQ | 605_CALCULATE_ETA_DAY | 22 | | | 22 | 7 | | | 7 | 3 | | | 3 | 114048 | | | 114048 |
| YGADWQ | 605_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGADWQ | 610_CHECK_LEAP_YEAR | 4 | | | 4 | 2 | | | 2 | 2 | | | 2 | 144 | | | 144 |
| YGADWQ | 610_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGADWQ | 615_ADD_TO_WORKING_DAY | 7 | | | 7 | 4 | | | 4 | 3 | | | 3 | 63 | | | 63 |
| YGADWQ | 615_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGADWQ | 620_SUBTRACT_FROM_WORKING_DA | 11 | | | 11 | 6 | | | 6 | 4 | | | 4 | 99 | | | 99 |
| YGADWQ | 620_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGAOWQ | 100_OPEN | 40 | | | | 8 | | | | 3 | | | | 1324960 | | | |
| YGAOWQ | 320_SEND_DATA | 11 | | | | 3 | | | | 3 | | | | 4851 | | | |
| YGAOWQ | 330_CLOSE_MESSAGE | 16 | | | | 4 | | | | 4 | | | | 28224 | | | |
| YGAOWQ | 400_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGAOWQ | 600_BAD_CODE | 4 | | | | 2 | | | | 2 | | | | 256 | | | |
| YGAOWQ | 650_CLOSE_MESSAGES | 12 | | | | 3 | | | | 3 | | | | 15552 | | | |
| YGAOWQ | 690_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGBTWQ | 001_START | 65 | | | 65 | 7 | | | 7 | 2 | | | 2 | 4394000 | | | 4394000 |
| YGBTWQ | 002_ABORT | 4 | | | 4 | 2 | | | 2 | 2 | | | 2 | 400 | | | 400 |
| YGBTWQ | 003_COUNTERS | 45 | | | 46 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGBTWQ | 003_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGBTWQ | 004_COUNTERS | 16 | | | 16 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGBTWQ | 004_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGBTWQ | 005_AF_COUNTERS | 13 | | | 13 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGBTWQ | 005_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGBTWQ | 006_ERR_COUNTERS | 2 | | | 2 | 1 | | | 2 | 1 | | | 1 | 8 | | | 72 |
| YGBTWQ | 006_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGBTWQ | 006A_ERR_DISP | 6 | | | 6 | 1 | | | 1 | 1 | | | 1 | 2400 | | | 3750 |
| YGBTWQ | 006A_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGCCWQ | 000_INITIALIZE | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCCWQ | 010_START | 6 | | | | 2 | | | | 2 | | | | 1536 | | | |
| YGCCWQ | 020_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGCCWQ | 100_BEGIN | 8 | | | | 2 | | | | 2 | | | | 6272 | | | |
| YGCCWQ | 110_LOOP | 26 | | | | 4 | | | | 2 | | | | 1269866 | | | |
| YGCCWQ | 140_PROCESS_OUT | 6 | | | | 3 | | | | 2 | | | | 24 | | | |

## MACA Subsystem Raw Product measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest Lvl | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April |
| YGCCWQ | 150_WRAP_UP | 3 | | | | 2 | | | | 1 | | | | 12 | | | |
| YGCCWQ | 160_ERROR_LOOK_UP | 6 | | | | 2 | | | | 2 | | | | 6144 | | | |
| YGCCWQ | 170_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGDGWQ | 100_START | 18 | | | 18 | 2 | | | 2 | 2 | | | 2 | 285768 | | | 285768 |
| YGDGWQ | 200_CYCLE | 7 | | | 7 | 3 | | | 3 | 2 | | | 2 | 28 | | | 28 |
| YGDGWQ | 300_ADD | 6 | | | 6 | 2 | | | 2 | 2 | | | 2 | 96 | | | 96 |
| YGDGWQ | 400_JD | 2 | | | 2 | 1 | | | 1 | 1 | | | 1 | 2 | | | 2 |
| YGDGWQ | 500_EXIT_PROGRAM | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 000_INITIALIZE | 3 | | | 3 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 010_DRIVER_PARA | 36 | | | 36 | 2 | | | 2 | 2 | | | 2 | 147456 | | | 147456 |
| YGEDWQ | 020_EXIT_PROGRAM | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 100_DOCUMENT_IDENTIFIER | 10 | | | 10 | 15 | | | 15 | 2 | | | 2 | 4000 | | | 4000 |
| YGEDWQ | 100_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 106_BAY_LOC | 2 | | | 2 | 1 | | | 1 | 1 | | | 1 | 2 | | | 2 |
| YGEDWQ | 106_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 108_HR_DATE_RCVD | 2 | | | 2 | 1 | | | 1 | 1 | | | 1 | 2 | | | 2 |
| YGEDWQ | 108_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 110_CONSIGNOR | 12 | | | 12 | 14 | | | 14 | 2 | | | 2 | 15552 | | | 15552 |
| YGEDWQ | 110_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 115_HR_DATE_SHPD | 2 | | | 2 | 1 | | | 1 | 1 | | | 1 | 2 | | | 2 |
| YGEDWQ | 115_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 120_AIR_CMDTY | 14 | | | 14 | 12 | | | 12 | 3 | | | 3 | 5600 | | | 5600 |
| YGEDWQ | 120_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 130_SP_HNDL | 19 | | | 19 | 22 | | | 21 | 3 | | | 3 | 17100 | | | 17100 |
| YGEDWQ | 130_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 140_AIR_DIMEN_CODE | 6 | | | 6 | 5 | | | 5 | 2 | | | 2 | 864 | | | 864 |
| YGEDWQ | 140_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 150_APOE | 8 | | | 8 | 12 | | | 12 | 2 | | | 2 | 1800 | | | 1800 |
| YGEDWQ | 150_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 160_APOD | 6 | | | 6 | 2 | | | 2 | 2 | | | 2 | 864 | | | 864 |
| YGEDWQ | 160_SEARCH | 16 | | | 16 | 5 | | | 5 | 3 | | | 3 | 14400 | | | 14400 |
| YGEDWQ | 160_ERROR | 4 | | | 4 | 1 | | | 1 | 1 | | | 1 | 36 | | | 36 |
| YGEDWQ | 160_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 170_SHPMT_MODE | 3 | | | 3 | 10 | | | 10 | 2 | | | 2 | 12 | | | 12 |
| YGEDWQ | 170_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 175_MANIF_REF | 2 | | | 2 | 1 | | | 1 | 1 | | | 1 | 2 | | | 2 |
| YGEDWQ | 175_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 180_PERS_PROPERTY_TCN_EDIT | 7 | | | 7 | 8 | | | 8 | 2 | | | 2 | 1575 | | | 1575 |
| YGEDWQ | 180_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 181_MOB_TCN_EDIT | 27 | | | 27 | 11 | | | 11 | 2 | | | 2 | 369603 | | | 369603 |
| YGEDWQ | 181_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 190_SAM_EDIT | 29 | | | 29 | 21 | | | 21 | 2 | | | 2 | 789525 | | | 789525 |
| YGEDWQ | 190_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 200_REGULAR_EDIT | 51 | | | 51 | 34 | | | 34 | 3 | | | 3 | 7364400 | | | 7364400 |
| YGEDWQ | 200_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 210_CONSIGNEE | 11 | | | 11 | 14 | | | 14 | 3 | | | 3 | 14256 | | | 14256 |
| YGEDWQ | 210_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 220_TRNSP_PRI_CRGO | 9 | | | 9 | 5 | | | 5 | 2 | | | 2 | 36 | | | 36 |
| YGEDWQ | 220_EDIT_PRI | 26 | | | 30 | 57 | | | 52 | 6 | | | 2 | 14976 | | | 9720 |
| YGEDWQ | 220_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 230_REQD_DELVRY_DA | 14 | | | 14 | 17 | | | 17 | 4 | | | 4 | 18144 | | | 18144 |
| YGEDWQ | 230_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 231_RDD_ERROR | 3 | | | 3 | 1 | | | 1 | 1 | | | 1 | 12 | | | 12 |
| YGEDWQ | 232_RDD_ERROR | 4 | | | 4 | 1 | | | 1 | 1 | | | 1 | 576 | | | 576 |
| YGEDWQ | 232_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 235_TEMP_EDIT_CORR | 8 | | | 8 | 4 | | | 4 | 3 | | | 3 | 5000 | | | 5000 |
| YGEDWQ | 235_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 240_HR_DATE_PROCD | 6 | | | 6 | 6 | | | 6 | 3 | | | 3 | 864 | | | 864 |
| YGEDWQ | 240_DATE | 5 | | | 5 | 3 | | | 3 | 2 | | | 2 | 180 | | | 180 |
| YGEDWQ | 240_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 250_ETA | 6 | | | 6 | 6 | | | 6 | 3 | | | 3 | 486 | | | 486 |
| YGEDWQ | 250_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |

# MACA Subsystem Raw Product measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest Lvl | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April |
| YGEDWQ | 260_TAC_CODE | 22 | | | 22 | 19 | | | 19 | 3 | | | 3 | 55000 | | | 55000 |
| YGEDWQ | 260_TAC_SPACES | 6 | | | 6 | 7 | | | 7 | 2 | | | 2 | 216 | | | 216 |
| YGEDWQ | 260_TAC_ALPHA | 29 | | | 29 | 15 | | | 15 | 3 | | | 3 | 190269 | | | 190269 |
| YGEDWQ | 260_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 270_PRIM_DOC_PIECES | 5 | | | 5 | 5 | | | 5 | 3 | | | 3 | 45 | | | 45 |
| YGEDWQ | 270_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 271_PCS_ERROR | 4 | | | 4 | 1 | | | 1 | 1 | | | 1 | 144 | | | 144 |
| YGEDWQ | 271_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 280_PRIM_DOC_WT | 12 | | | 12 | 6 | | | 6 | 3 | | | 3 | 4800 | | | 4800 |
| YGEDWQ | 280_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 281_WT_ERROR | 4 | | | 4 | 1 | | | 1 | 1 | | | 1 | 324 | | | 324 |
| YGEDWQ | 281_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 282_WT_ERROR | 4 | | | 4 | 1 | | | 1 | 1 | | | 1 | 144 | | | 144 |
| YGEDWQ | 282_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 290_PRIM_DOC_CUBE | 12 | | | 12 | 6 | | | 6 | 3 | | | 3 | 4800 | | | 4800 |
| YGEDWQ | 290_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 291_CUBE_ERROR | 4 | | | 4 | 1 | | | 1 | 1 | | | 1 | 324 | | | 324 |
| YGEDWQ | 291_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 292_CUBE_ERROR | 4 | | | 4 | 1 | | | 1 | 1 | | | 1 | 144 | | | 144 |
| YGEDWQ | 292_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 300_LIST_ERRORS | 6 | | | 12 | 2 | | | 4 | 2 | | | 3 | 486 | | | 5292 |
| YGEDWQ | 301_DEFAULT_MSG | 16 | | | 24 | 2 | | | 4 | 2 | | | 2 | 230400 | | | 470400 |
| YGEDWQ | 302_BEGIN_LISTING | 7 | | | 7 | 2 | | | 2 | 2 | | | 2 | 63 | | | 63 |
| YGEDWQ | 303_WRITE | 7 | | | 15 | 1 | | | 3 | 1 | | | 2 | 1008 | | | 8640 |
| YGEDWQ | 303_A_EXIT_PROGRAM | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 304_WRITE_ERR_MESSAGE | 2 | | | 2 | 1 | | | 1 | 1 | | | 1 | 2 | | | 2 |
| YGEDWQ | 305_MOVES | 60 | | | 76 | 9 | | | 13 | 3 | | | 3 | 3810240 | | | 6569136 |
| YGEDWQ | 305_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 400_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 500_ERROR_COUNT | 21 | | | 21 | 23 | | | 23 | 6 | | | 6 | 27216 | | | 27216 |
| YGEDWQ | 500_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 510_AF_COUNT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 16 | | | 16 |
| YGEDWQ | 510_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 520_AM_COUNT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 16 | | | 16 |
| YGEDWQ | 520_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 530_NAV_COUNT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 36 | | | 36 |
| YGEDWQ | 530_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 540_MAR_COUNT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 16 | | | 16 |
| YGEDWQ | 540_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 900_FILL_TABLE | 10 | | | 16 | 2 | | | 4 | 2 | | | 4 | 4199040 | | | 7683984 |
| YGEDWQ | 900_FIND_DEVICE | 4 | | | 4 | 4 | | | 4 | 2 | | | 2 | 16 | | | 16 |
| YGEDWQ | 900_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 1000_TOTAL_COUNTERS | 72 | | | 72 | 25 | | | 25 | 3 | | | 3 | 373248 | | | 373248 |
| YGEDWQ | 1000_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 1100_VALID_COUNTERS | 82 | | | 83 | 28 | | | 29 | 3 | | | 3 | 1845000 | | | 2689200 |
| YGEDWQ | 1100_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 1200_INVALID_COUNTERS | 82 | | | 82 | 28 | | | 28 | 3 | | | 3 | 1260832 | | | 1260832 |
| YGEDWQ | 1200_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 1301_AF_CHECK_COUNT | 10 | | | 10 | 12 | | | 12 | 5 | | | 5 | 360 | | | 360 |
| YGEDWQ | 1301_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGEDWQ | 1302_TCN_1_CHECK | 9 | | | 9 | 15 | | | 15 | 5 | | | 5 | 81 | | | 81 |
| YGEDWQ | 1302_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGETWQ | 010_HOUSEKEEPING | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 020_DRIVER_PARA | 36 | | | | 13 | | | | 10 | | | | 1440000 | | | |
| YGETWQ | 030_LIST_ERRORS | 18 | | | | 2 | | | | 2 | | | | 490050 | | | |
| YGETWQ | 050_BEGIN_LISTING | 7 | | | | 1 | | | | 1 | | | | 1575 | | | |
| YGETWQ | 060_WRITE_ERR_MESSAGE | 1 | | | | 1 | | | | 1 | | | | 1 | | | |
| YGETWQ | 070_MOVES | 37 | | | | 5 | | | | 2 | | | | 1480000 | | | |
| YGETWQ | 070_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 090_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 100_TE6_EDIT | 0 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGETWQ | 110_ROUND_COUNT | 5 | | | | 3 | | | | 2 | | | | 180 | | | |

# MACA Subsystem Raw Product measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest Lvl | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April |
| YGETWQ | 120_LOAD_STORAGE_GROUP | 4 | | | | 2 | | | | 2 | | | | 144 | | | |
| YGETWQ | 130_STOCK_NUMBER | 20 | | | | 8 | | | | 6 | | | | 48020 | | | |
| YGETWQ | 140_DODIC | 9 | | | | 10 | | | | 3 | | | | 5184 | | | |
| YGETWQ | 150_CLASSIFICATION | 20 | | | | 10 | | | | 2 | | | | 118580 | | | |
| YGETWQ | 190_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 200_TJ6_EDIT | 0 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGETWQ | 210_ROUND_COUNT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 220_LOAD_STORAGE_GROUP | 4 | | | | 2 | | | | 2 | | | | 144 | | | |
| YGETWQ | 230_STOCK_NUMBER | 20 | | | | 8 | | | | 6 | | | | 48020 | | | |
| YGETWQ | 240_DODIC | 4 | | | | 3 | | | | 2 | | | | 144 | | | |
| YGETWQ | 250_COMPARABILITY_CODE | 4 | | | | 2 | | | | 2 | | | | 144 | | | |
| YGETWQ | 290_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 300_TE7_EDIT | 0 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGETWQ | 310_NEW | 4 | | | | 3 | | | | 2 | | | | 144 | | | |
| YGETWQ | 320_EXPLOSIVE_LOT_NBR | 2 | | | | 1 | | | | 1 | | | | 8 | | | |
| YGETWQ | 320_CONT | 9 | | | | 4 | | | | 2 | | | | 5625 | | | |
| YGETWQ | 330_PIECES | 4 | | | | 3 | | | | 2 | | | | 144 | | | |
| YGETWQ | 340_WEIGHT | 4 | | | | 3 | | | | 2 | | | | 144 | | | |
| YGETWQ | 350_CUBE | 4 | | | | 3 | | | | 2 | | | | 144 | | | |
| YGETWQ | 390_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 400_T9_EDIT | 0 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGETWQ | 410_SUPPLEMENTAL_INFO | 9 | | | | 2 | | | | 2 | | | | 3969 | | | |
| YGETWQ | 420_SEQUENCE_NUMBER | 6 | | | | 2 | | | | 2 | | | | 216 | | | |
| YGETWQ | 490_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 500_TH8_TF8_EDIT | 0 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGETWQ | 510_CONSIGNOR | 4 | | | | 2 | | | | 2 | | | | 144 | | | |
| YGETWQ | 520_LAST_NAME | 8 | | | | 2 | | | | 2 | | | | 2592 | | | |
| YGETWQ | 520_CONT | 9 | | | | 9 | | | | 2 | | | | 5625 | | | |
| YGETWQ | 530_INITIALS | 4 | | | | 2 | | | | 2 | | | | 144 | | | |
| YGETWQ | 540_GRADE | 8 | | | | 5 | | | | 3 | | | | 1800 | | | |
| YGETWQ | 550_POS_71 | 4 | | | | 2 | | | | 2 | | | | 144 | | | |
| YGETWQ | 560_WEIGHT | 6 | | | | 4 | | | | 2 | | | | 486 | | | |
| YGETWQ | 570_POS_77_80 | 4 | | | | 2 | | | | 2 | | | | 144 | | | |
| YGETWQ | 590_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 600_TP8_EDIT | 0 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGETWQ | 610_POV_YEAR | 4 | | | | 2 | | | | 2 | | | | 144 | | | |
| YGETWQ | 620_POV_MAKE | 4 | | | | 3 | | | | 2 | | | | 144 | | | |
| YGETWQ | 630_TP8_LAST_NAME | 5 | | | | 3 | | | | 2 | | | | 720 | | | |
| YGETWQ | 640_TP8_INITIAL | 4 | | | | 3 | | | | 2 | | | | 144 | | | |
| YGETWQ | 650_TP8_GRADE | 8 | | | | 5 | | | | 3 | | | | 1800 | | | |
| YGETWQ | 660_VEH_LIC_PLATE_STATE | 4 | | | | 4 | | | | 2 | | | | 576 | | | |
| YGETWQ | 670_VEH_LIC_PLATE_NO | 9 | | | | 11 | | | | 6 | | | | 2916 | | | |
| YGETWQ | 680_VEH_COLOR_DESCRIPTION | 4 | | | | 3 | | | | 2 | | | | 144 | | | |
| YGETWQ | 690_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 700_T5_EDIT | 0 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGETWQ | 710_CONSIGNOR | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 720_T_5_LENGTH | 5 | | | | 3 | | | | 2 | | | | 320 | | | |
| YGETWQ | 730_T_5_WIDTH | 5 | | | | 3 | | | | 2 | | | | 320 | | | |
| YGETWQ | 740_T_5_HEIGHT | 5 | | | | 3 | | | | 2 | | | | 320 | | | |
| YGETWQ | 750_T_5_PIECES | 11 | | | | 5 | | | | 3 | | | | 2816 | | | |
| YGETWQ | 760_T_5_WEIGHT | 11 | | | | 5 | | | | 3 | | | | 2816 | | | |
| YGETWQ | 770_T_5_CUBE | 11 | | | | 5 | | | | 3 | | | | 2816 | | | |
| YGETWQ | 790_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 800_TV5_EDIT | 0 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGETWQ | 810_MODEL | 7 | | | | 2 | | | | 2 | | | | 700 | | | |
| YGETWQ | 810_CONT | 8 | | | | 4 | | | | 2 | | | | 2048 | | | |
| YGETWQ | 820_BII_FIELD | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 830_BII_NUMBER | 4 | | | | 2 | | | | 2 | | | | 144 | | | |
| YGETWQ | 840_CARGO_LENGTH | 5 | | | | 3 | | | | 2 | | | | 320 | | | |
| YGETWQ | 850_CARGO_WIDTH | 5 | | | | 3 | | | | 2 | | | | 320 | | | |
| YGETWQ | 860_CARGO_HEIGHT | 5 | | | | 3 | | | | 2 | | | | 320 | | | |
| YGETWQ | 870_SERIAL_NUMBER | 2 | | | | 1 | | | | 1 | | | | 8 | | | |

## MACA Subsystem Raw Product measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest Lvl | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April |
| YGETWQ | 870_CONT | 9 | | | | 4 | | | | 2 | | | | 5625 | | | |
| YGETWQ | 890_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 900_FILL_TABLE | 8 | | | | 2 | | | | 2 | | | | 3548448 | | | |
| YGETWQ | 910_FIND_DEVICE | 3 | | | | 4 | | | | 2 | | | | 48 | | | |
| YGETWQ | 910_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 1000_VALID_COUNTERS | 18 | | | | 13 | | | | 10 | | | | 23328 | | | |
| YGETWQ | 1000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 1100_INVALID_COUNTERS | 18 | | | | 13 | | | | 10 | | | | 23328 | | | |
| YGETWQ | 1100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 1200_T_6_EDIT | 0 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGETWQ | 1210_ROUND_COUNT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGETWQ | 1230_STOCK_NUMBER | 20 | | | | 8 | | | | 6 | | | | 48020 | | | |
| YGETWQ | 1240_NOMENCLATURE | 7 | | | | 3 | | | | 2 | | | | 567 | | | |
| YGETWQ | 1290_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGFPWQ | 000_DRIVER | 26 | | | | 4 | | | | 2 | | | | 988650 | | | |
| YGFPWQ | 100_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGFPWQ | 200_PURGE_DRIVER | 33 | | | | 9 | | | | 3 | | | | 1069200 | | | |
| YGFPWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGFPWQ | 300_DEL_DRIVER | 9 | | | | 7 | | | | 2 | | | | 2304 | | | |
| YGFPWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGFPWQ | 400_DEL_BY_ADDRS | 5 | | | | 2 | | | | 2 | | | | 1280 | | | |
| YGFPWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGFPWQ | 500_DEL_BY_KEYVALUE | 5 | | | | 2 | | | | 2 | | | | 1280 | | | |
| YGFPWQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGFPWQ | 600_RELEASE_REC | 5 | | | | 1 | | | | 1 | | | | 180 | | | |
| YGFPWQ | 600_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGFPWQ | 700_STARS_RESULT | 102 | | | | 45 | | | | 2 | | | | 1619352 | | | |
| YGFPWQ | 700_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGFPWQ | 800_CALL_YG12WQ | 2 | | | | 1 | | | | 1 | | | | 648 | | | |
| YGFPWQ | 800_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGFPWQ | 900_RTOS_GONE | 2 | | | | 1 | | | | 1 | | | | 32 | | | |
| YGFPWQ | 900_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGFPWQ | 1000_OPS_CONSOLE | 7 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGFPWQ | 1000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGFPWQ | 1301_AF_CHECK_COUNT | 10 | | | | 13 | | | | 6 | | | | 360 | | | |
| YGFPWQ | 1301_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGFPWQ | 1302_TCN_1_CHECK | 7 | | | | 14 | | | | 4 | | | | 28 | | | |
| YGFPWQ | 1302_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGINWQ | 000_START | 56 | | | | 18 | | | | 7 | | | | 8257536 | | | |
| YGINWQ | 000_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGINWQ | 100_CALL_RCOMC | 47 | | | | 17 | | | | 3 | | | | 5306112 | | | |
| YGINWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGINWQ | 200_MINIS_BREAKDOWN | 6 | | | | 2 | | | | 2 | | | | 3750 | | | |
| YGINWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGINWQ | 300_CRT_BREAKDOWN | 7 | | | | 4 | | | | 3 | | | | 567 | | | |
| YGINWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGINWQ | 400_INIT_STARS | 10 | | | | 3 | | | | 2 | | | | 12250 | | | |
| YGINWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGINWQ | 500_INIT_RBTCHO | 13 | | | | 3 | | | | 2 | | | | 29952 | | | |
| YGINWQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGINWQ | 600_EOO_CALL | 10 | | | | 3 | | | | 2 | | | | 20250 | | | |
| YGINWQ | 600_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGINWQ | 650_SEND_NULL | 17 | | | | 3 | | | | 2 | | | | 156672 | | | |
| YGINWQ | 650_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGINWQ | 700_SEND_NULL | 14 | | | | 3 | | | | 2 | | | | 222264 | | | |
| YGINWQ | 700_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGINWQ | 800_APOE_BLOCK | 4 | | | | 1 | | | | 1 | | | | 256 | | | |
| YGINWQ | 810_MOVE_KEY | 15 | | | | 4 | | | | 2 | | | | 77760 | | | |
| YGINWQ | 800_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMAWQ | 100_BEGIN | 6 | | | | 1 | | | | 1 | | | | 3456 | | | |
| YGMAWQ | 300_PROCESS | 98 | | | | 47 | | | | 6 | | | | 219325568 | | | |
| YGMAWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

## MACA Subsystem Raw Product measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest Lvl | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April |
| YGMAWQ | 400_CHECK_RZLT | 15 | | | | 5 | | | | 5 | | | | 24000 | | | |
| YGMAWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMAWQ | 500_SAVE | 9 | | | | 2 | | | | 2 | | | | 0 | | | |
| YGMAWQ | 500_CALL_SUBSAVE | 15 | | | | 4 | | | | 2 | | | | 13500 | | | |
| YGMAWQ | 500_SLEEP | 1 | | | | 1 | | | | 1 | | | | 36 | | | |
| YGMAWQ | 500_SLEEP_EXIT | 0 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGMAWQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMOWQ | 000_CONTROL_MODULE | 24 | | | | 10 | | | | 2 | | | | 912600 | | | |
| YGMOWQ | 000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMOWQ | 100_INIT_UNIT_BLOCK | 32 | | | | 2 | | | | 2 | | | | 6027392 | | | |
| YGMOWQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMOWQ | 200_SEND_UNIT_BLOCKS | 44 | | | | 8 | | | | 2 | | | | 13310000 | | | |
| YGMOWQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMOWQ | 300_BUILD_UNIT_BLOCKS | 35 | | | | 19 | | | | 10 | | | | 591500 | | | |
| YGMOWQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMOWQ | 340_SETUP_TO_CLEAR_UB | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMOWQ | 340_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMOWQ | 350_WRITE_UB_REC | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMOWQ | 350_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMOWQ | 400_ADD_570 | 8 | | | | 1 | | | | 1 | | | | 18432 | | | |
| YGMOWQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMOWQ | 500_ADD_580 | 9 | | | | 3 | | | | 2 | | | | 28224 | | | |
| YGMOWQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMOWQ | 1000_CONVERT_TIME | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMOWQ | 1000_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMOWQ | 1100_TIME_CHECK | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMOWQ | 1100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMOWQ | 1301_AF_CHECK_COUNT | 10 | | | | 13 | | | | 6 | | | | 640 | | | |
| YGMOWQ | 1301_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGMOWQ | 1302_TCN_1_CHECK | 9 | | | | 15 | | | | 5 | | | | 81 | | | |
| YGMOWQ | 1302_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGOTWQ | 001_START | 21 | | | 21 | 5 | | | 5 | 2 | | | 2 | 27216 | | | 27216 |
| YGOTWQ | 100_SEND | 24 | | | 25 | 25 | | | 25 | 6 | | | 6 | 7776 | | | 14400 |
| YGOTWQ | 200_EXIT_PROGRAM | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGOTWQ | 300_MINIS_OUTPUT | 30 | | | 30 | 4 | | | 4 | 2 | | | 2 | 1310430 | | | 1310430 |
| YGOTWQ | 300_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGOTWQ | 500_RBTCHO_CRT_OUTPUT | 35 | | | 35 | 6 | | | 6 | 3 | | | 3 | 1819440 | | | 1819440 |
| YGOTWQ | 500_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 000_START | 33 | | | 41 | 7 | | | 10 | 4 | | | 5 | 1597200 | | | 4628736 |
| YGRNWQ | 000_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 100_BUILD_ADVANCE | 33 | | | 40 | 8 | | | 11 | 2 | | | 2 | 5765892 | | | 11924640 |
| YGRNWQ | 100_ADD_ADVANCE | 12 | | | 22 | 4 | | | 8 | 3 | | | 3 | 10800 | | | 95832 |
| YGRNWQ | 100_BUILD_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 110_CHECK_ZONE | 10 | | | 10 | 6 | | | 6 | 2 | | | 2 | 16000 | | | 16000 |
| YGRNWQ | 110_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 115_APO_SEARCH | 7 | | | 7 | 4 | | | 4 | 2 | | | 2 | 4032 | | | 4032 |
| YGRNWQ | 115_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 200_NO_HIT | 38 | | | 58 | 11 | | | 18 | 3 | | | 3 | 3558168 | | | 12272800 |
| YGRNWQ | 200_NO_HIT_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 300_UPDATE_ADVANCE | 14 | | | 17 | 3 | | | 4 | 2 | | | 2 | 201600 | | | 347633 |
| YGRNWQ | 300_REPLACE | 13 | | | 13 | 4 | | | 4 | 3 | | | 3 | 11700 | | | 11700 |
| YGRNWQ | 300_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 500_SEND_NOTICE | 17 | | | 17 | 6 | | | 6 | 4 | | | 4 | 34425 | | | 34425 |
| YGRNWQ | 500_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 510_SERVICE_CHECK | 9 | | | 9 | 19 | | | 19 | 5 | | | 5 | 2025 | | | 2025 |
| YGRNWQ | 510_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 513_AIR_FORCE | 6 | | | 6 | 2 | | | 2 | 2 | | | 2 | 4704 | | | 4704 |
| YGRNWQ | 513_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 514_ARMY | 6 | | | 6 | 2 | | | 2 | 2 | | | 2 | 7350 | | | 7350 |
| YGRNWQ | 514_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 515_NAVY | 6 | | | 6 | 2 | | | 2 | 2 | | | 2 | 7350 | | | 7350 |
| YGRNWQ | 515_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |

## MACA Subsystem Raw Product measures

| Program Name | Paragraph (Procedure) Name | # of Stmts | | | | McCabe's CC | | | | Max Nest Lvl | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April |
| YGRNWQ | 516_MARINE | 9 | | | 9 | 2 | | | 2 | 2 | | | 2 | 14400 | | | 14400 |
| YGRNWQ | 516_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 517_GET_ADDRESS | 12 | | | 12 | 3 | | | 3 | 2 | | | 2 | 52272 | | | 52272 |
| YGRNWQ | 517_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 610_SERVICE_CHECK | 13 | | | 13 | 24 | | | 24 | 6 | | | 6 | 22932 | | | 22932 |
| YGRNWQ | 610_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 700_DODAAC_CHECK | 9 | | | 9 | 14 | | | 14 | 4 | | | 4 | 2025 | | | 2025 |
| YGRNWQ | 700_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 710_DODAAC_COUNT | 7 | | | 7 | 14 | | | 14 | 4 | | | 4 | 1008 | | | 1008 |
| YGRNWQ | 710_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 800_COUNTER | 6 | | | 6 | 8 | | | 8 | 3 | | | 3 | 1536 | | | 1536 |
| YGRNWQ | 800_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 810_COUNTER | 9 | | | 9 | 15 | | | 15 | 5 | | | 5 | 1296 | | | 1296 |
| YGRNWQ | 810_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 900_YGAOWQ_CALL | 17 | | | 17 | 2 | | | 2 | 2 | | | 2 | 108800 | | | 108800 |
| YGRNWQ | 900_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| YGRNWQ | 910_PROJ_CODE_CHK | | | | 3 | | | | 3 | | | | 2 | | | | 12 |
| YGRNWQ | 910_EXIT | | | | 1 | | | | 1 | | | | 1 | | | | 0 |
| YGRNWQ | 920_CODE_J_CHK | | | | 4 | | | | 6 | | | | 2 | | | | 784 |
| YGRNWQ | 920_EXIT | | | | 1 | | | | 1 | | | | 1 | | | | 0 |
| YGS4WQ | 000_START | 6 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS4WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS4WQ | 100_INIT_HDR8 | 1 | | | | 1 | | | | 1 | | | | 324 | | | |
| YGS4WQ | 199_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS4WQ | 200_INIT_HDR10 | 1 | | | | 1 | | | | 1 | | | | 729 | | | |
| YGS4WQ | 299_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS4WQ | 300_REPORT | 3 | | | | 1 | | | | 1 | | | | 27 | | | |
| YGS4WQ | 310_READ | 37 | | | | 9 | | | | 3 | | | | 3601728 | | | |
| YGS4WQ | 310_CONT | 20 | | | | 5 | | | | 3 | | | | 196020 | | | |
| YGS4WQ | 399_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS4WQ | 310_SERVICE_CHANGE | 11 | | | | 4 | | | | 1 | | | | 43659 | | | |
| YGS4WQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS4WQ | 320_CONSIGNOR_CHANGE | 8 | | | | 3 | | | | 2 | | | | 25088 | | | |
| YGS4WQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS4WQ | 400_SERVICE_REPORT | 11 | | | | 6 | | | | 6 | | | | 99 | | | |
| YGS4WQ | 499_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS4WQ | 500_WRITE_HEADER | 10 | | | | 9 | | | | 1 | | | | 4840 | | | |
| YGS4WQ | 599_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS4WQ | 600_INIT_HEADER | 2 | | | | 1 | | | | 1 | | | | 162 | | | |
| YGS4WQ | 699_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS4WQ | 700_SAVE_HEADER | 12 | | | | 1 | | | | 1 | | | | 292032 | | | |
| YGS4WQ | 799_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS4WQ | 800_WRITE_SUMMARY | 38 | | | | 34 | | | | 1 | | | | 196992 | | | |
| YGS4WQ | 899_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS5WQ | 000_DRIVER | 17 | | | | 7 | | | | 2 | | | | 5508 | | | |
| YGS5WQ | 000_EXIT_PROGRAM | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS5WQ | 100_SLEW_TRANSLATOR_HSKP | 13 | | | | 3 | | | | 2 | | | | 1300 | | | |
| YGS5WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS5WQ | 200_SLEW_TRANSLATOR_DRIVER | 4 | | | | 2 | | | | 2 | | | | 64 | | | |
| YGS5WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS5WQ | 210_READ_REPORT_FILE | 6 | | | | 3 | | | | 2 | | | | 384 | | | |
| YGS5WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS5WQ | 220_TRANSLATE_AND_WRITE | 42 | | | | 9 | | | | 2 | | | | 122472 | | | |
| YGS5WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS5WQ | 221_SPACE_LINE | 5 | | | | 2 | | | | 1 | | | | 2880 | | | |
| YGS5WQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS5WQ | 222_REPORT_DATA | 4 | | | | 3 | | | | 2 | | | | 1936 | | | |
| YGS5WQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS5WQ | 300_SLEW_TRANSLATOR_WRAPUP | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS5WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS5WQ | 400_RPT_TRANSMITTER_HSKP | 18 | | | | 1 | | | | 1 | | | | 152352 | | | |
| YGS5WQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

## MACA Subsystem Raw Product measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest Lvl | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April |
| YGS5WQ | 500_RPT_TRANSMITTER_DRIVER | 4 | | | | 3 | | | | 1 | | | | 256 | | | |
| YGS5WQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS5WQ | 520_BD700_SEND_DRIVER | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| YGS5WQ | 520_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS5WQ | 521_FIRST_READ | 10 | | | | 2 | | | | 1 | | | | 3240 | | | |
| YGS5WQ | 521_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS5WQ | 522_READ_SEND_RECORD | 10 | | | | 3 | | | | 2 | | | | 10240 | | | |
| YGS5WQ | 522_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS5WQ | 700_ERROR_DISPLAY | 6 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGS5WQ | 700_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX2WQ | 000_START | 9 | | | | 3 | | | | 2 | | | | 81 | | | |
| YGX2WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX2WQ | 100_BOJ | 13 | | | | 2 | | | | 2 | | | | 40768 | | | |
| YGX2WQ | 199_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX2WQ | 200_BUILD | 15 | | | | 4 | | | | 2 | | | | 116160 | | | |
| YGX2WQ | 205_CALL | 63 | | | | 34 | | | | 7 | | | | 40118652 | | | |
| YGX2WQ | 205_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX2WQ | 210_SERVICE_CHOICES | 6 | | | | 3 | | | | 2 | | | | 864 | | | |
| YGX2WQ | 300_WRAP_UP | 7 | | | | 2 | | | | 2 | | | | 2268 | | | |
| YGX2WQ | 399_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX2WQ | 400_SERVICE_CHECK | 12 | | | | 20 | | | | 6 | | | | 432 | | | |
| YGX2WQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX2WQ | 410_SERVICE_CHECK | 15 | | | | 25 | | | | 7 | | | | 6000 | | | |
| YGX2WQ | 410_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX2WQ | 420_DODAAC_CHECK | 8 | | | | 14 | | | | 4 | | | | 32 | | | |
| YGX2WQ | 420_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX2WQ | 440_DLA | 13 | | | | 6 | | | | 6 | | | | 832 | | | |
| YGX2WQ | 440_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX2WQ | 500_SET_RANGE | 17 | | | | 5 | | | | 2 | | | | 49572 | | | |
| YGX2WQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX3WQ | 000_START | 3 | | | | 2 | | | | 2 | | | | 147 | | | |
| YGX3WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 000_START | 17 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 100_INIT_HDR8 | 1 | | | | 1 | | | | 1 | | | | 225 | | | |
| YGX4WQ | 199_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 200_INIT_HDR10 | 1 | | | | 1 | | | | 1 | | | | 256 | | | |
| YGX4WQ | 299_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 300_REPORT | 5 | | | | 1 | | | | 1 | | | | 125 | | | |
| YGX4WQ | 310_READ | 40 | | | | 12 | | | | 3 | | | | 7225000 | | | |
| YGX4WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 310_CHANGE_SERVICE | 15 | | | | 7 | | | | 3 | | | | 77760 | | | |
| YGX4WQ | 310_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 320_CONSIGNOR_CHANGE | 19 | | | | 6 | | | | 2 | | | | 462384 | | | |
| YGX4WQ | 320_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 330_FINISH_HEADER8 | 15 | | | | 5 | | | | 2 | | | | 47040 | | | |
| YGX4WQ | 330_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 335_WRITE_HDR8 | 9 | | | | 3 | | | | 3 | | | | 5625 | | | |
| YGX4WQ | 335_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 400_SERVICE_REPORT | 22 | | | | 6 | | | | 6 | | | | 12672 | | | |
| YGX4WQ | 499_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 500_WRITE_HEADER | 33 | | | | 4 | | | | 3 | | | | 363825 | | | |
| YGX4WQ | 599_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 600_INIT_HEADER | 1 | | | | 1 | | | | 1 | | | | 9 | | | |
| YGX4WQ | 699_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 700_SAVE_HEADER | 15 | | | | 1 | | | | 1 | | | | 91260 | | | |
| YGX4WQ | 799_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 800_WRITE_SUMMARY | 44 | | | | 30 | | | | 2 | | | | 372416 | | | |
| YGX4WQ | 899_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 900_DLA_BREAKDOWN | 14 | | | | 3 | | | | 1 | | | | 72576 | | | |
| YGX4WQ | 900_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 920_DLA_SERVICE | 11 | | | | 6 | | | | 6 | | | | 99 | | | |

MACA Subsystem Raw Product measures

| | | # of Stmts | | | | McCabe's CC | | | | Max Nest Lvl | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April | Baseline | October | January | March/April |
| YGX4WQ | 920_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 930_SAVE_HEADER | 11 | | | | 1 | | | | 1 | | | | 39600 | | | |
| YGX4WQ | 930_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX4WQ | 950_AF_OVERSEAS_COSIGNORS | 16 | | | | 4 | | | | 2 | | | | 147456 | | | |
| YGX4WQ | 950_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX5WQ | 000_DRIVER | 17 | | | | 7 | | | | 2 | | | | 5508 | | | |
| YGX5WQ | 000_STOP_RUN | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX5WQ | 100_SLEW_TRANSLATOR_HSKP | 13 | | | | 3 | | | | 2 | | | | 1300 | | | |
| YGX5WQ | 100_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX5WQ | 200_SLEW_TRANSLATOR_DRIVER | 4 | | | | 2 | | | | 2 | | | | 64 | | | |
| YGX5WQ | 200_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX5WQ | 210_READ_REPORT_FILE | 6 | | | | 3 | | | | 2 | | | | 384 | | | |
| YGX5WQ | 210_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX5WQ | 220_TRANSLATE_AND_WRITE | 42 | | | | 9 | | | | 2 | | | | 122472 | | | |
| YGX5WQ | 220_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX5WQ | 221_SPACE_LINE | 5 | | | | 2 | | | | 1 | | | | 2880 | | | |
| YGX5WQ | 221_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX5WQ | 222_REPORT_DATA | 4 | | | | 3 | | | | 2 | | | | 1936 | | | |
| YGX5WQ | 222_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX5WQ | 300_SLEW_TRANSLATOR_WRAPUP | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX5WQ | 300_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX5WQ | 400_RPT_TRANSMITTER_HSKP | 18 | | | | 1 | | | | 1 | | | | 152352 | | | |
| YGX5WQ | 400_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX5WQ | 500_RPT_TRANSMITTER_DRIVER | 4 | | | | 3 | | | | 1 | | | | 256 | | | |
| YGX5WQ | 500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX5WQ | 520_BD700_SEND_DRIVER | 2 | | | | 2 | | | | 1 | | | | 32 | | | |
| YGX5WQ | 520_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX5WQ | 521_FIRST_READ | 12 | | | | 3 | | | | 2 | | | | 6912 | | | |
| YGX5WQ | 521_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX5WQ | 522_READ_SEND_RECORD | 10 | | | | 3 | | | | 2 | | | | 10240 | | | |
| YGX5WQ | 522_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX5WQ | 700_ERROR_DISPLAY | 6 | | | | 1 | | | | 1 | | | | 0 | | | |
| YGX5WQ | 700_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

## CRQS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Nest Level | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | January | April | Baseline | October | January | April | Baseline | October | January | April | Baseline | October | January | April |
| AUTIN | 000_DRIVER | 6 | | | | 4 | | | | 1 | | | | 2400 | | | |
| AUTIN | 0100_READ_AUTODIN_MESG | 13 | | | | 5 | | | | 4 | | | | 15925 | | | |
| AUTIN | 0300_RAUTIN_ERR | 6 | | | | 5 | | | | 3 | | | | 0 | | | |
| BLDISP | 0100_DRIVER | 4 | | | | 2 | | | | 1 | | | | 36 | | | |
| BLDISP | 0200_OPEN_FILES | 5 | | | | 2 | | | | 1 | | | | 80 | | | |
| BLDISP | 0300_READ_INPUT | 1 | | | | 1 | | | | 1 | | | | 1 | | | |
| BLDISP | 0400_BUILD_KEY_GETRAN | 14 | | | | 3 | | | | 2 | | | | 43904 | | | |
| BLDISP | 0410_EDIT | 38 | | | | 19 | | | | 4 | | | | 2470950 | | | |
| BLDISP | 0411_CHK_PQ_CQ | 5 | | | | 3 | | | | 3 | | | | 180 | | | |
| BLDISP | 0412_CHK_HQ_DQ | 5 | | | | 3 | | | | 3 | | | | 180 | | | |
| BLDISP | 0413_CHK_BQ_TQ | 5 | | | | 3 | | | | 3 | | | | 180 | | | |
| BLDISP | 0414_CHK_MQ_AQ | 5 | | | | 3 | | | | 3 | | | | 180 | | | |
| BLDISP | 0420_BUILD_NUM_FIELDS | 26 | | | | 9 | | | | 3 | | | | 355914 | | | |
| BLDISP | 0430_CHGREC | 6 | | | | 2 | | | | 2 | | | | 54 | | | |
| BLDISP | 0440_NEXT_READ | 2 | | | | 2 | | | | 1 | | | | 2 | | | |
| BLDISP | 0500_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| BLDISP | 0600_CLOSE_FILES | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| BLDISP | 0700_CONUS_OUT_CHECK | 2 | | | | 2 | | | | 2 | | | | 32 | | | |
| BLDISP | 0900_GETRAN_MESSAGE | 7 | | | | 5 | | | | 2 | | | | 1008 | | | |
| BLDISP | 0900_ATL_PAC_MESSAGE | 10 | | | | 5 | | | | 2 | | | | 9000 | | | |
| BLDISP | 0900_INB_OUTB_MESSAGE | 10 | | | | 5 | | | | 2 | | | | 9000 | | | |
| BLDISP | 0900_SIGN_INCORRECT | 10 | | | | 5 | | | | 2 | | | | 23040 | | | |
| BLDISP | 0900_MONTH_MESSAGE | 8 | | | | 5 | | | | 2 | | | | 2048 | | | |
| BLDISP | 0900_CHANGE_NUMBER_MESSAG | 8 | | | | 5 | | | | 2 | | | | 2048 | | | |
| BLDISP | 0900_NONNUMERIC_MESSAGE | 8 | | | | 5 | | | | 2 | | | | 6272 | | | |
| BLDISP | 0900_TOP_OF_PAGE | 4 | | | | 2 | | | | 1 | | | | 3600 | | | |
| BLDISP | 0900_NON_CONUS_MESSAGE | 8 | | | | 5 | | | | 2 | | | | 2048 | | | |
| BLDISP | 0900_NON_CARGO_MESSAGE | 8 | | | | 5 | | | | 2 | | | | 2048 | | | |
| CRQINT | 0001_A_INITILIZE | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| CRQINT | 0001_SORT_INPUT_FILE | 3 | | | | 1 | | | | 1 | | | | 108 | | | |
| CRQINT | 0001_DRIVER | 4 | | | | 2 | | | | 1 | | | | 144 | | | |
| CRQINT | 0005_OPEN | 6 | | | | 3 | | | | 1 | | | | 54 | | | |
| CRQINT | 0005_RELEASE_INPUT_REC | 9 | | | | 2 | | | | 1 | | | | 20736 | | | |
| CRQINT | 0006_CARGO_PASS | 14 | | | | 3 | | | | 2 | | | | 896 | | | |
| CRQINT | 0010_SECTION_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| CRQINT | 0100_DRIVER | 11 | | | | 3 | | | | 1 | | | | 2816 | | | |
| CRQINT | 0100_OPEN | 5 | | | | 2 | | | | 1 | | | | 20 | | | |
| CRQINT | 0100_NOPEN | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| CRQINT | 0100_READ | 6 | | | | 3 | | | | 1 | | | | 864 | | | |
| CRQINT | 0100_ICLOSE | 3 | | | | 1 | | | | 1 | | | | 0 | | | |
| CRQINT | 0200_SECOND_SORT | 2 | | | | 1 | | | | 1 | | | | 32 | | | |
| CRQINT | 0200_OPEN | 5 | | | | 1 | | | | 1 | | | | 180 | | | |
| CRQINT | 0200_WRITE_DETAILS | 6 | | | | 3 | | | | 2 | | | | 384 | | | |
| CRQINT | 205_PRINT_LINE_FORMAT | 21 | | | | 8 | | | | 2 | | | | 108864 | | | |
| CRQINT | 0200_CLOSE | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| CRQINT | 0500_PRINT_HEADER | 3 | | | | 2 | | | | 1 | | | | 48 | | | |
| CRQINT | 010_SECTION_EXIT | 0 | | | | 1 | | | | 0 | | | | 0 | | | |
| NEWQUO | 0001_MAIN_INIT | 12 | | | | 1 | | | | 1 | | | | 71148 | | | |
| NEWQUO | 0010_CHN_STORE | 8 | | | | 3 | | | | 2 | | | | 128 | | | |
| NEWQUO | 0011_CHN_END | 2 | | | | 1 | | | | 1 | | | | 32 | | | |
| NEWQUO | 0020_CARD_READ | 10 | | | | 3 | | | | 2 | | | | 4000 | | | |
| NEWQUO | 0021_CONUS_CHECK | 24 | | | | 4 | | | | 2 | | | | 194400 | | | |
| NEWQUO | 0060_INTRA | 6 | | | | 1 | | | | 1 | | | | 3750 | | | |
| NEWQUO | 0065_TABLE_STORE | 2 | | | | 2 | | | | 2 | | | | 0 | | | |
| NEWQUO | 0070_SORT_LOOP | 18 | | | | 10 | | | | 6 | | | | 0 | | | |
| NEWQUO | 0073_LOOP_TEST | 4 | | | | 2 | | | | 2 | | | | 16 | | | |
| NEWQUO | 0075_STORE | 8 | | | | 2 | | | | 2 | | | | 1800 | | | |
| NEWQUO | 0076_STORE | 30 | | | | 9 | | | | 5 | | | | 1323000 | | | |
| NEWQUO | 0080_QUOTA_STORE | 2 | | | | 1 | | | | 1 | | | | 8 | | | |

# CRQS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Nest Level | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | January | April | Baseline | October | January | April | Baseline | October | January | April | Baseline | October | January | April |
| NEWQUO | 0081_QUOTA_LOOP | 6 | | | | 2 | | | | 2 | | | | 864 | | | |
| NEWQUO | 090_GETSEQ | 17 | | | | 4 | | | | 2 | | | | 100793 | | | |
| NEWQUO | 090_GETSEQ_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NEWQUO | 091_FIRST_FIELD | 3 | | | | 2 | | | | 2 | | | | 48 | | | |
| NEWQUO | 092_2ND_FIELD | 3 | | | | 2 | | | | 2 | | | | 48 | | | |
| NEWQUO | 093_3RD_FIELD | 3 | | | | 2 | | | | 2 | | | | 48 | | | |
| NEWQUO | 094_4TH_FIELD | 3 | | | | 2 | | | | 2 | | | | 48 | | | |
| NEWQUO | 095_OTHER_MOVES | 9 | | | | 2 | | | | 2 | | | | 28224 | | | |
| NEWQUO | 0100_PROCESS | 11 | | | | 1 | | | | 1 | | | | 85184 | | | |
| NEWQUO | 0105_REPEAT | 9 | | | | 1 | | | | 1 | | | | 1296 | | | |
| NEWQUO | 0106_REPEAT | 8 | | | | 1 | | | | 1 | | | | 4608 | | | |
| NEWQUO | 0108_DOUBLE_REPEAT | 18 | | | | 3 | | | | 2 | | | | 139392 | | | |
| NEWQUO | 0110_STOP | 5 | | | | 2 | | | | 1 | | | | 5 | | | |
| NEWQUO | 0501_BEGIN | 2 | | | | 2 | | | | 2 | | | | 0 | | | |
| NEWQUO | 0505_NEXT_SUB_CHN | 9 | | | | 3 | | | | 2 | | | | 5184 | | | |
| NEWQUO | 0505_CGO | 40 | | | | 5 | | | | 4 | | | | 20391840 | | | |
| NEWQUO | 0507_CHECK_AIR_FORCE | 8 | | | | 3 | | | | 3 | | | | 800 | | | |
| NEWQUO | 0508_END_SUB_CHN | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NEWQUO | 0510_TOTAL | 3 | | | | 1 | | | | 1 | | | | 12 | | | |
| NEWQUO | 0510_LOOP | 15 | | | | 2 | | | | 2 | | | | 105840 | | | |
| NEWQUO | 0512_PRINT | 5 | | | | 1 | | | | 1 | | | | 80 | | | |
| NEWQUO | 0520_END | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NEWQUO | 0600_LAST_CHAN | 3 | | | | 1 | | | | 1 | | | | 0 | | | |
| NEWQUO | 0610_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NEWQUO | 0530_BEGIN | 5 | | | | 2 | | | | 2 | | | | 45 | | | |
| NEWQUO | 0530_LOOP | 11 | | | | 2 | | | | 2 | | | | 43659 | | | |
| NEWQUO | 0532_PRINT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NEWQUO | 0533_CLEAR | 2 | | | | 1 | | | | 1 | | | | 0 | | | |
| NEWQUO | 0533_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NEWQUO | 0910_TOP_OF_PAGE | 8 | | | | 3 | | | | 1 | | | | 1152 | | | |
| NEWQUO | 0910_CONT | 3 | | | | 3 | | | | 1 | | | | 12 | | | |
| NEWQUO | 0911_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NEWQUO | 0920_PRINT_2 | 7 | | | | 3 | | | | 2 | | | | 63 | | | |
| NEWQUO | 0921_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NEWQUO | 0940_BOT_OF_PAGE | 3 | | | | 2 | | | | 1 | | | | 48 | | | |
| NEWQUO | 0941_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NEWQUO | 0950_PRINT_1 | 7 | | | | 3 | | | | 2 | | | | 63 | | | |
| NEWQUO | 0951_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NEWQUO | 9998_TOTSEC | 8 | | | | 2 | | | | 2 | | | | 1800 | | | |
| NEWQUO | 9998_START | 10 | | | | 2 | | | | 2 | | | | 2250 | | | |
| NEWQUO | 9998_CHECK_STN | 7 | | | | 3 | | | | 2 | | | | 448 | | | |
| NEWQUO | 9998_START_1 | 1 | | | | 1 | | | | 1 | | | | 1 | | | |
| NEWQUO | 9998_SIFT | 13 | | | | 5 | | | | 2 | | | | 40768 | | | |
| NEWQUO | 9998_SIFT1 | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NEWQUO | 9998_PRT_INTRA | 4 | | | | 1 | | | | 1 | | | | 0 | | | |
| NEWQUO | 9998_TOTS | 5 | | | | 2 | | | | 2 | | | | 180 | | | |
| NEWQUO | 9998_END | 10 | | | | 4 | | | | 2 | | | | 4000 | | | |
| NEWQUO | 9998_GRAND_TOTAL | 10 | | | | 3 | | | | 2 | | | | 17640 | | | |
| NEWQUO | 9998_NEXT | 5 | | | | 2 | | | | 2 | | | | 45 | | | |
| NEWQUO | 9998_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NEWQUO | 9998_A_CHECK_FOR_NUMBERS | 1 | | | | 2 | | | | 1 | | | | 4 | | | |
| NEWQUO | 9998_A_EXIT | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NEWQUO | 9998_B_CHECK_FOR_NUMBERS | 4 | | | | 3 | | | | 2 | | | | 100 | | | |
| NEWQUO | 9998_B_EXIT | 0 | | | | 1 | | | | 0 | | | | 0 | | | |
| NEWQUO | 9999_INPT1 | 17 | | | | 8 | | | | 4 | | | | 3825 | | | |
| NEWQUO | 9999_EXIT1 | 1 | | | | 1 | | | | 1 | | | | 0 | | | |
| NEWQUO | 9999_OUPT1 | 2 | | | | 1 | | | | 1 | | | | 2 | | | |
| NEWQUO | 9999_OUPT2 | 5 | | | | 1 | | | | 1 | | | | 20 | | | |
| NEWQUO | 9999_EXIT2 | 1 | | | | 1 | | | | 1 | | | | 0 | | | |

## CRQS Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Nest Level | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | January | April | Baseline | October | January | April | Baseline | October | January | April | Baseline | October | January | April |
| CRQS_UP | 000_INIT | 47 | | | 48 | 10 | | | 10 | 4 | | | 4 | 33957500 | | | 36750000 |
| CRQS_UP | 010_INTRO | 27 | | | 27 | 3 | | | 3 | 2 | | | 2 | 2430000 | | | 2430000 |
| CRQS_UP | 020_MENU | 58 | | | 58 | 11 | | | 11 | 2 | | | 2 | 2509312 | | | 2509312 |
| CRQS_UP | 025_REPORT | 11 | | | 11 | 3 | | | 3 | 1 | | | 1 | 6875 | | | 6875 |
| CRQS_UP | 030_ADD | 21 | | | 21 | 4 | | | 4 | 1 | | | 1 | 1281189 | | | 1281189 |
| CRQS_UP | 030_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| CRQS_UP | 035_ADDITION | 22 | | | 22 | 4 | | | 4 | 4 | | | 4 | 371800 | | | 371800 |
| CRQS_UP | 035A_REWRITE | 7 | | | 7 | 1 | | | 1 | 1 | | | 1 | 7 | | | 7 |
| CRQS_UP | 036_CARGO_ADDS | 30 | | | 30 | 5 | | | 5 | 2 | | | 2 | 768000 | | | 768000 |
| CRQS_UP | 038_CHECK_EXIST | 8 | | | 8 | 9 | | | 9 | 5 | | | 5 | 288 | | | 288 |
| CRQS_UP | 040_DELETE | 28 | | | 28 | 5 | | | 5 | 2 | | | 2 | 5347132 | | | 5347132 |
| CRQS_UP | 040_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| CRQS_UP | 041_DELETION | 9 | | | 9 | 2 | | | 2 | 2 | | | 2 | 44100 | | | 44100 |
| CRQS_UP | 041A_REWRITE | 7 | | | 7 | 1 | | | 1 | 1 | | | 1 | 7 | | | 7 |
| CRQS_UP | 050_UPDATE | 41 | | | 41 | 7 | | | 7 | 3 | | | 3 | 15206121 | | | 15206121 |
| CRQS_UP | 050_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| CRQS_UP | 051_CHANGES | 28 | | | 28 | 4 | | | 4 | 2 | | | 2 | 1211392 | | | 1211392 |
| CRQS_UP | 053_CARGO_CHANGES | 27 | | | 27 | 7 | | | 7 | 2 | | | 2 | 1179387 | | | 1179387 |
| CRQS_UP | 057_NEW_VALUE | 9 | | | 9 | 1 | | | 1 | 1 | | | 1 | 15876 | | | 15876 |
| CRQS_UP | 060_DISPLAY | 21 | | | 21 | 3 | | | 3 | 1 | | | 1 | 435456 | | | 435456 |
| CRQS_UP | 060_RESET_ISP_FILE | 2 | | | 2 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| CRQS_UP | 060_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| CRQS_UP | 061_GETSEQ | 10 | | | 10 | 5 | | | 5 | 3 | | | 3 | 20250 | | | 20250 |
| CRQS_UP | 061_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| CRQS_UP | 063_CARGO_MOVES | 16 | | | 16 | 2 | | | 2 | 1 | | | 1 | 608400 | | | 608400 |
| CRQS_UP | 064_DISP_ON_SCREEN | 17 | | | 17 | 1 | | | 1 | 1 | | | 1 | 100793 | | | 100793 |
| CRQS_UP | 065_TAPE | 62 | | | 62 | 7 | | | 7 | 4 | | | 4 | 16252928 | | | 16252928 |
| CRQS_UP | 066_LOAD_TABLE | 3 | | | 3 | 2 | | | 2 | 1 | | | 1 | 48 | | | 48 |
| CRQS_UP | 067_WRITE_CARD | 2 | | | 2 | 2 | | | 2 | 1 | | | 1 | 8 | | | 8 |
| CRQS_UP | 069_FIND_IN_TABLE | 6 | | | 6 | 2 | | | 2 | 2 | | | 2 | 864 | | | 864 |
| CRQS_UP | 070_CREATE_MONTH | 66 | | | 66 | 7 | | | 7 | 4 | | | 4 | 46569600 | | | 46569600 |
| CRQS_UP | 075_LOAD_TABLE | 3 | | | 3 | 2 | | | 2 | 1 | | | 1 | 48 | | | 48 |
| CRQS_UP | 076_FIND_NEW_MON | 5 | | | 5 | 3 | | | 3 | 3 | | | 3 | 320 | | | 320 |
| CRQS_UP | 077_WRITE_PARAM | 2 | | | 2 | 2 | | | 2 | 1 | | | 1 | 8 | | | 8 |
| CRQS_UP | 080_VALID_SERV | 10 | | | 10 | 7 | | | 7 | 2 | | | 2 | 51840 | | | 51840 |
| CRQS_UP | 085_VALID_REQ | 3 | | | 3 | 1 | | | 1 | 1 | | | 1 | 108 | | | 108 |
| CRQS_UP | 090_CHOOSE_TAPE | 26 | | | 26 | 6 | | | 6 | 4 | | | 4 | 958464 | | | 958464 |
| CRQS_UP | 095_SEND_ADANS | 9 | | | 9 | 2 | | | 2 | 1 | | | 1 | 5184 | | | 5184 |
| CRQS_UP | 100_READ_DISK | 11 | | | 11 | 4 | | | 4 | 3 | | | 3 | 34496 | | | 34496 |
| CRQS_UP | 100_START | 4 | | | 4 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| CRQS_UP | 120_MOVE_WRITE | 20 | | | 20 | 3 | | | 3 | 2 | | | 2 | 1132880 | | | 1132880 |
| CRQS_UP | 140_SEND_AMPS | 28 | | | 28 | 3 | | | 3 | 2 | | | 2 | 444528 | | | 444528 |
| CRQS_UP | 140_SEND_AMPS_EXIT | 1 | | | 1 | 1 | | | 1 | 1 | | | 1 | 0 | | | 0 |
| CRQS_UP | 900_GETRAN_MESS | 10 | | | 10 | 1 | | | 1 | 1 | | | 1 | 64000 | | | 64000 |
| CRQS_UP | 901_DISPLAY_MESS | 8 | | | 8 | 1 | | | 1 | 1 | | | 1 | 9800 | | | 9800 |
| CRQS_UP | 910_SERV_ERR | 9 | | | 9 | 1 | | | 1 | 1 | | | 1 | 28224 | | | 28224 |
| CRQS_UP | 920_REQ_ERR | 9 | | | 9 | 1 | | | 1 | 1 | | | 1 | 11025 | | | 11025 |
| CRQS_UP | 930_VALID_DATAB | 13 | | | 13 | 2 | | | 2 | 2 | | | 2 | 316368 | | | 316368 |
| CRQS_UP | 940_TAPE_ERR | 9 | | | 9 | 1 | | | 1 | 1 | | | 1 | 20736 | | | 20736 |
| CRQS_UP | 950_ACCESS_WORKFILES | 21 | | | 21 | 1 | | | 1 | 1 | | | 1 | 287469 | | | 287469 |

Over Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Nest Level | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | January | April | Baseline | October | January | April | Baseline | October | January | April | Baseline | October | January | April |
| YGOSWQ | 000_INITIALIZE | 11 | | | | 3 | | | | 1 | | | | 2816 | | | |
| YGOSWQ | 010_START_PROCESSING | 32 | | | | 10 | | | | 2 | | | | 259200 | | | |
| YGOSWQ | 010_BYPASS_MESSAGE | 21 | | | | 10 | | | | 2 | | | | 42525 | | | |
| YGOSWQ | 010_CONTINUE | 6 | | | | 2 | | | | 1 | | | | 96 | | | |
| YGOSWQ | 020_STARS_CALL | 54 | | | | 9 | | | | 3 | | | | 6316056 | | | |
| YGOSWQ | 020_EXIT | 1 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGOSWQ | 030_BUILD_HOLD_AREA | 14 | | | | 4 | | | | 1 | | | | 12600 | | | |
| YGOSWQ | 030_CONT | 10 | | | | 5 | | | | 1 | | | | 4000 | | | |
| YGOSWQ | 030_EXIT | 1 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGOSWQ | 070_SEND_CT_MESSAGES | 23 | | | | 6 | | | | 1 | | | | 94208 | | | |
| YGOSWQ | 070_EXIT | 1 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGOSWQ | 100_SEND_BCHDIN_BUFFER | 19 | | | | 2 | | | | 1 | | | | 55404 | | | |
| YGOSWQ | 100_EXIT | 1 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGOSWQ | 200_SEND_PLAS | 7 | | | | 2 | | | | 1 | | | | 1008 | | | |
| YGOSWQ | 210_CONT | 7 | | | | 2 | | | | 1 | | | | 567 | | | |
| YGOSWQ | 220_SEND | 12 | | | | 2 | | | | 1 | | | | 21168 | | | |
| YGOSWQ | 230_EXIT | 1 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGOSWQ | 410_SEND_CT_DATA | 16 | | | | 3 | | | | 1 | | | | 123904 | | | |
| YGOSWQ | 410_EXIT | 1 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGOSWQ | 500_SEND_CLOSE_MESSAGE | 11 | | | | 2 | | | | 1 | | | | 26411 | | | |
| YGOSWQ | 500_EXIT | 1 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGOSWQ | 600_SYSTEM_ID_CALL | 21 | | | | 2 | | | | 1 | | | | 254100 | | | |
| YGOSWQ | 600_EXIT | 1 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGOSWQ | 800_RTOS_I_CALL | 19 | | | | 6 | | | | 1 | | | | 346275 | | | |
| YGOSWQ | 810_NULL_O_CALL | 11 | | | | 2 | | | | 1 | | | | 57024 | | | |
| YGOSWQ | 815_BOJ_CALL | 11 | | | | 2 | | | | 1 | | | | 32076 | | | |
| YGOSWQ | 910_EOO_CALL | 16 | | | | 3 | | | | 1 | | | | 30976 | | | |
| YGOSWQ | 1000_STOP_RUN | 2 | | | | 1 | | | | 0 | | | | 32 | | | |

Over Subsystem Raw Product Measures

| | | # of Stmts | | | | McCabe's CC | | | | Nest Level | | | | Information Flow | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Program Name | Paragraph (Procedure) Name | Baseline | October | January | April | Baseline | October | January | April | Baseline | October | January | April | Baseline | October | January | April |
| YGOSWQ | 000_INITIALIZE | 11 | | | | 3 | | | | 1 | | | | 2816 | | | |
| YGOSWQ | 010_START_PROCESSING | 32 | | | | 10 | | | | 2 | | | | 259200 | | | |
| YGOSWQ | 010_BYPASS_MESSAGE | 21 | | | | 10 | | | | 2 | | | | 42525 | | | |
| YGOSWQ | 010_CONTINUE | 6 | | | | 2 | | | | 1 | | | | 96 | | | |
| YGOSWQ | 020_STARS_CALL | 54 | | | | 9 | | | | 3 | | | | 6316056 | | | |
| YGOSWQ | 020_EXIT | 1 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGOSWQ | 030_BUILD_HOLD_AREA | 14 | | | | 4 | | | | 1 | | | | 12600 | | | |
| YGOSWQ | 030_CONT | 10 | | | | 5 | | | | 1 | | | | 4000 | | | |
| YGOSWQ | 030_EXIT | 1 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGOSWQ | 070_SEND_CT_MESSAGES | 23 | | | | 6 | | | | 1 | | | | 94208 | | | |
| YGOSWQ | 070_EXIT | 1 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGOSWQ | 100_SEND_BCHDIN_BUFFER | 19 | | | | 2 | | | | 1 | | | | 55404 | | | |
| YGOSWQ | 100_EXIT | 1 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGOSWQ | 200_SEND_PLAS | 7 | | | | 2 | | | | 1 | | | | 1008 | | | |
| YGOSWQ | 210_CONT | 7 | | | | 2 | | | | 1 | | | | 567 | | | |
| YGOSWQ | 220_SEND | 12 | | | | 2 | | | | 1 | | | | 21168 | | | |
| YGOSWQ | 230_EXIT | 1 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGOSWQ | 410_SEND_CT_DATA | 16 | | | | 3 | | | | 1 | | | | 123904 | | | |
| YGOSWQ | 410_EXIT | 1 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGOSWQ | 500_SEND_CLOSE_MESSAGE | 11 | | | | 2 | | | | 1 | | | | 26411 | | | |
| YGOSWQ | 500_EXIT | 1 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGOSWQ | 600_SYSTEM_ID_CALL | 21 | | | | 2 | | | | 1 | | | | 254100 | | | |
| YGOSWQ | 600_EXIT | 1 | | | | 1 | | | | 0 | | | | 0 | | | |
| YGOSWQ | 800_RTOS_I_CALL | 19 | | | | 6 | | | | 1 | | | | 346275 | | | |
| YGOSWQ | 810_NULL_O_CALL | 11 | | | | 2 | | | | 1 | | | | 57024 | | | |
| YGOSWQ | 815_BOJ_CALL | 11 | | | | 2 | | | | 1 | | | | 32076 | | | |
| YGOSWQ | 910_EOO_CALL | 16 | | | | 3 | | | | 1 | | | | 30976 | | | |
| YGOSWQ | 1000_STOP_RUN | 2 | | | | 1 | | | | 0 | | | | 32 | | | |

# *Appendix C*

Defect Data from AMC/CSS

HOST Defects by Release

| | DEVELOPMENT PHASE | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Analysis | | | | | | Prelim Design | | | | | | Detail Design | | | | | | Coding | | | | | | Unit Test | | | | | | Formal Test | | | | | | |
| Change Request by Release | Priority 1 | Priority 2 | Priority 3 | Priority 4 | Priority 5 | Sub-Total | Priority 1 | Priority 2 | Priority 3 | Priority 4 | Priority 5 | Sub-Total | Priority 1 | Priority 2 | Priority 3 | Priority 4 | Priority 5 | Sub-Total | Priority 1 | Priority 2 | Priority 3 | Priority 4 | Priority 5 | Sub-Total | Priority 1 | Priority 2 | Priority 3 | Priority 4 | Priority 5 | Sub-Total | Priority 1 | Priority 2 | Priority 3 | Priority 4 | Priority 5 | Sub-Total | Grand Total |
| **October** | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| NXH-238 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | 3 | | 3 | 3 |
| NXH-252 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | 1 | | | 1 | | 2 | 2 |
| NXH-256 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | 4 | | 4 | 4 |
| NXH-258 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | 1 | | | | | 1 | 1 |
| NXH-263 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | 3 | | 3 | 3 |
| NXH-269 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | 1 | | 1 | 1 |
| YG3-178 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | 0 |
| **Sub-Total** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 12 | 0 | 14 | 14 |
| **January** | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| NXH-267 | | | 1 | | | 1 | | | | | | 0 | | | | 1 | | 1 | | | | | | 0 | | | | | | 0 | | | | | | 0 | 2 |
| NXH-271 | | | | | | 0 | | | | 2 | | 2 | | | 2 | | | 2 | | | 1 | | | 1 | | | | | | 0 | | | | | | 0 | 5 |
| *Formal Peer Reviews Begin* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| YG3-165 | | 2 | 2 | | | 4 | | | 3 | 2 | | 5 | 3 | | 5 | 4 | | 12 | 1 | | 1 | 1 | | 3 | 1 | | 2 | | | 3 | | | | | | 0 | 27 |
| YG5-135 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | 1 | | 1 | 1 |
| **Sub-Total** | 0 | 2 | 3 | 0 | 0 | 5 | 0 | 0 | 3 | 4 | 0 | 7 | 3 | 0 | 7 | 5 | 0 | 15 | 1 | 0 | 2 | 1 | 0 | 4 | 1 | 0 | 2 | 0 | 0 | 3 | 0 | 0 | 0 | 1 | 0 | 1 | 35 |
| **April** | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| NXH-274 | | | | 2 | | 2 | | | | | | 0 | | | | | | 0 | | | | | | 0 | 1 | | 1 | | | 2 | | | | | | 0 | 4 |
| CR1-20 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | 0 |
| YG1-207 | 2 | | 2 | | | 4 | | | 1 | 1 | | 2 | 2 | | | 6 | | 8 | 4 | 4 | 4 | 6 | | 18 | | | | | | 0 | | | | | | 0 | 32 |
| YG1-208 | | | | 1 | | 1 | 1 | | | 1 | | 2 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | 3 |
| YG3-163 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | | | | | | 0 | 0 |
| YG3-179 | | | | | | 0 | 2 | | 1 | 1 | | 4 | 1 | | | 1 | | 2 | | | | | | 0 | | | | | | 0 | | | | | | 0 | 6 |
| **Sub-Total** | 2 | 0 | 2 | 3 | 0 | 7 | 3 | 0 | 2 | 3 | 0 | 8 | 3 | 0 | 0 | 7 | 0 | 10 | 4 | 4 | 4 | 6 | 0 | 18 | 1 | 0 | 1 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 45 |
| **Grand Total** | 2 | 2 | 5 | 3 | 0 | 12 | 3 | 0 | 5 | 7 | 0 | 15 | 6 | 0 | 7 | 12 | 0 | 25 | 5 | 4 | 6 | 7 | 0 | 22 | 2 | 0 | 3 | 0 | 0 | 5 | 2 | 0 | 0 | 13 | 0 | 15 | 94 |

# Process and Product Measures by Change Request and Program

| Release | Subsystem | Change Number | Defects | File | * = New Program | Manhours | Total Procedures | Total Stmts | Largest Proc | Avg Stmts/proc | Total CC | Max CC | Avg CC | Max NL | Avg NL | Total IF | Max IF | Avg IF |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Oct | T | NXH-252 | 2 | NXR7WQ | | 8 | 5 | 36 | 9 | 7 | 12 | 4 | 2 | 2 | 2 | 12960 | 5832 | 2592 |
| | T | NXH-256 | 4 | NXICWQ | | 1 | 21 | 374 | 107 | 18 | 125 | 20 | 6 | 2 | 2 | 45827835 | 40336800 | 2182278 |
| | | | | NXIDWQ | | 4 | 39 | 435 | 45 | 11 | 183 | 19 | 5 | 2 | 2 | 32589413 | 21458944 | 814735 |
| | T | NXH-269 | 1 | NXMHWQ | | 27 | 29 | 624 | 92 | 22 | 184 | 13 | 6 | 3 | 2 | 378116858 | 248349308 | 13038512 |
| | T | NXH-263 | 3 | NXPMWQ | | 1 | 28 | 165 | 46 | 12 | 35 | 5 | 3 | 2 | 2 | 12545021 | 12105774 | 896073 |
| | | | | NXRGWQ | | 1 | 33 | 383 | 40 | 12 | 146 | 12 | 4 | 3 | 2 | 9950499 | 5416960 | 301530 |
| | | | | NXPYWQ | | 1 | 16 | 326 | 91 | 20 | 102 | 16 | 6 | 3 | 2 | 690654368 | 685687275 | 43165898 |
| | | | | NXRPWQ | | 1 | 16 | 146 | 25 | 9 | 61 | 7 | 4 | 2 | 2 | 789107 | 608400 | 49319 |
| | | | | NXU1WQ | | 1 | 38 | 684 | 100 | 18 | 320 | 25 | 8 | 3 | 2 | 153777927 | 116208400 | 4046788 |
| | T | NXH-238 | 3 | NXD0WQ | * | 8 | 4 | 12 | 5 | 3 | 7 | 3 | 2 | 1 | 1 | 1152 | 1125 | 288 |
| | | | | NXD1WQ | * | 8 | 5 | 24 | 5 | 5 | 12 | 3 | 2 | 2 | 1 | 2626 | 1125 | 525 |
| | | | | NXD2WQ | * | 8 | 7 | 52 | 10 | 7 | 21 | 4 | 3 | 2 | 2 | 41292 | 12960 | 5899 |
| | | | | NXD3WQ | * | 8 | 5 | 25 | 6 | 5 | 12 | 3 | 2 | 2 | 1 | 3072 | 1350 | 614 |
| | | | | NXD4WQ | * | 8 | 3 | 12 | 6 | 4 | 6 | 3 | 2 | 1 | 1 | 1530 | 1350 | 510 |
| | | | | NXD5WQ | * | 8 | 3 | 12 | 6 | 4 | 6 | 3 | 2 | 1 | 1 | 1530 | 1350 | 510 |
| | | | | NXD7WQ | * | 8 | 4 | 12 | 5 | 3 | 7 | 3 | 2 | 1 | 1 | 1152 | 1125 | 288 |
| | | | | NXIEWQ | * | 175 | 15 | 220 | 38 | 15 | 81 | 19 | 5 | 2 | 2 | 46292137 | 30711638 | 3086142 |
| | U | YG3-178 | 0 | YG15WQ | | 1 | 16 | 519 | 87 | 32 | 195 | 51 | 12 | 3 | 2 | 54193695 | 43118592 | 3387106 |
| | | | | YGABWQ | | 5 | 18 | 306 | 56 | 17 | 80 | 11 | 4 | 4 | 2 | 113744450 | 99058400 | 6319136 |
| | | | | YGALWQ | | 10 | 33 | 669 | 73 | 21 | 206 | 21 | 6 | 4 | 2 | 365044311 | 177652800 | 11407635 |
| | | | | YGLFWQ | | 10 | 35 | 676 | 68 | 19 | 210 | 21 | 6 | 4 | 2 | 287908338 | 157313988 | 8225953 |
| | | | | YGOBWQ | | 5 | 19 | 532 | 156 | 28 | 211 | 83 | 11 | 3 | 2 | 2766801947 | 2751840000 | 145621155 |
| | | | | YGORWQ | | 5 | 21 | 303 | 50 | 14 | 100 | 12 | 5 | 4 | 2 | 2694783 | 1427712 | 128323 |
| | | | | YGPFWQ | | 5 | 10 | 171 | 71 | 17 | 61 | 32 | 6 | 2 | 2 | 5118565 | 3973888 | 511857 |
| | | | | | | | | | | | | | | | | | | |
| Jan | T | NXH-271 | 5 | NXMBWQ | | 20 | 18 | 285 | 60 | 17 | 69 | 15 | 4 | 2 | 1 | 46975611 | 25947000 | 2763271 |
| | T | NXH-259 | 3 | NXIFWQ | * | 66 | 12 | 116 | 32 | 10 | 47 | 11 | 4 | 2 | 2 | 15456570 | 15235200 | 1288048 |
| | U | YG3-165 | 27 | YGDRWQ | | 8 | 26 | 376 | 93 | 14 | 88 | 16 | 3 | 4 | 2 | 241613163 | 223432500 | 9292814 |
| | | | | YGCTWQ | * | 250 | 25 | 287 | 73 | 11 | 69 | 21 | 3 | 5 | 2 | 10503955 | 9513433 | 420158 |
| | | | | YGUCWQ | | 10 | 24 | 493 | 159 | 21 | 140 | 42 | 6 | 5 | 2 | 1912992736 | 1887015975 | 79708031 |
| | | | | | | | | | | | | | | | | | | |
| Apr | M | YG1-211 | 0 | YGDGWQ | | 2 | 4 | 33 | 18 | 8 | 8 | 3 | 2 | 2 | 2 | 285894 | 285768 | 71474 |
| | M | YG1-206 | 1 | YGRNWQ | | 50 | | | | | | | | | | | | |
| | M | YG1-208 | 3 | YGRNWQ | | 8 | | | | | | | | | | | | |
| | M | YG1-209 | 1 | YGRNWQ | | 8 | | | | | | | | | | | | |
| | M | YG1-207 | 32 | YGRNWQ | | 30 | 23 | 286 | 38 | 14 | 160 | 24 | 8 | 6 | 3 | 11425515 | 5765892 | 544072 |
| | | | | YGADWQ | | 40 | 27 | 416 | 103 | 15 | 169 | 35 | 6 | 5 | 2 | 5288988 | 3276800 | 195888 |
| | | | | YGEDWQ | | 10 | 59 | 849 | 82 | 14 | 514 | 57 | 9 | 6 | 2 | 20791611 | 7364400 | 352400 |
| | | | | YGBTWQ | | 5 | 7 | 151 | 65 | 22 | 14 | 7 | 2 | 2 | 1 | 4396808 | 4394000 | 628115 |
| | | | | YGOTWQ | | 5 | 4 | 110 | 35 | 28 | 40 | 25 | 10 | 6 | 3 | 3164862 | 1819440 | 791216 |
| | U | YG3-179 | 6 | YGCTWQ | | 50 | 27 | 366 | 76 | 14 | 98 | 18 | 4 | 5 | 2 | 22431502 | 19305216 | 418019 |
| | T | NXH-274 | 4 | NXM1WQ | | 5 | 9 | 51 | 9 | 6 | 27 | 8 | 3 | 2 | 1 | 11300 | 4375 | 1256 |
| | | | | NXMPWQ | | 5 | 20 | 421 | 54 | 21 | 156 | 24 | 8 | 4 | 2 | 54997739 | 29292984 | 2749887 |

# *Appendix E*

## Product Measures by Changed Modules

| File Name | * = New Programs | Module/Procedure Name | Pre-Stmts | Post-Stmts | Δ Stmts | Stmts Affected | Pre-McCabe | Post-McCabe | Δ McCabe | Pre-Nest Level | Post-Nest Level | Δ Nest Level | Pre-Info Flow | Post-Info Flow | Δ Info Flow |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| NXR7WQ | | 000_CLOSE | 8 | 9 | 1 | 1 | 2 | 2 | 0 | 2 | 2 | 0 | 5832 | 8100 | 2268 |
| | | 500_TCN_CHECK | 9 | 10 | 1 | 1 | 4 | 4 | 0 | 2 | 2 | 0 | 2304 | 4000 | 1696 |
| | | **TOTAL** | **17** | **19** | **2** | **2** | **6** | **6** | **0** | **4** | **4** | **0** | **8136** | **12100** | **3964** |
| NXICWQ | | 100_HSKP | 42 | 43 | 1 | 1 | 9 | 9 | 0 | 2 | 2 | 0 | 40336800 | 44299675 | 3962875 |
| | | 700_SELECT_AND_MOVE | 20 | 18 | -2 | 4 | 20 | 19 | -1 | 2 | 2 | 0 | 208080 | 145800 | -62280 |
| | | **TOTAL** | **62** | **61** | **-1** | **5** | **29** | **28** | **-1** | **4** | **4** | **0** | **40544880** | **44445475** | **3900595** |
| NXIDWQ | | 110_SELECT | 23 | 19 | -4 | 4 | 6 | 6 | 0 | 2 | 2 | 0 | 1195632 | 109744 | -1085888 |
| | | 720_PROCESS | 32 | 32 | 0 | 1 | 19 | 18 | -1 | 2 | 2 | 0 | 4333568 | 3612672 | -720896 |
| | | **TOTAL** | **55** | **51** | **-4** | **5** | **25** | **24** | **-1** | **4** | **4** | **0** | **5529200** | **3722416** | **-1806784** |
| NXMHWQ | | 000_CONTROL | 15 | 16 | 1 | 2 | 2 | 2 | 0 | 2 | 2 | 0 | 24000 | 32400 | 8400 |
| | | 131_SELECT_INPUT | 27 | 30 | 3 | 3 | 13 | 14 | 1 | 2 | 2 | 0 | 456300 | 711480 | 255180 |
| | | 420_AIR_TOTALS | 92 | 117 | 25 | 25 | 13 | 17 | 4 | 2 | 2 | 0 | 248349308 | 635726637 | 387377329 |
| | | 440_TRK_TOTALS | 58 | 81 | 23 | 23 | 9 | 11 | 2 | 2 | 2 | 0 | 51795450 | 170772624 | 118977174 |
| | | **TOTAL** | **192** | **244** | **52** | **53** | **37** | **44** | **7** | **8** | **8** | **0** | **300625058** | **807243141** | **506618083** |
| NXPMWQ | | 211_ALLOCATE_YESTERDAY_FILE | 5 | 11 | 6 | 6 | 5 | 8 | 3 | 2 | 2 | 0 | 3645 | 39600 | 35955 |
| NXRGWQ | | 220_PROCESS | 27 | 32 | 5 | 7 | 12 | 15 | 3 | 2 | 2 | 0 | 1179387 | 2230272 | 1050885 |
| NXPYWQ | | 200_REFORMAT_INPUT_FILE | 17 | 20 | 3 | 6 | 7 | 9 | 2 | 2 | 2 | 0 | 88128 | 162000 | 73872 |
| NXRPWQ | | 131_ADDFIL_CALL | 8 | 13 | 5 | 7 | 5 | 8 | 3 | 2 | 2 | 0 | 24200 | 83200 | 59000 |
| NXU1WQ | | 210_ALLOCATE_AKF_FILE | 19 | 29 | 10 | 10 | 11 | 14 | 3 | 2 | 2 | 0 | 134064 | 1160000 | 1025936 |
| NXD0WQ | * | 000_DRIVER | | 3 | | | | 2 | | | 1 | | | 27 | |
| | * | 100_HSKP | | 1 | | | | 1 | | | 1 | | | 0 | |
| | * | 200_READ | | 5 | | | | 3 | | | 1 | | | 1125 | |
| | * | 300_WRAPUP | | 3 | | | | 1 | | | 1 | | | 0 | |
| | * | **TOTAL** | | **12** | | | | **7** | | | **4** | | | **1152** | |
| NXD1WQ | * | 000_DRIVER | | 4 | | | | 2 | | | 2 | | | 16 | |
| | * | 100_ONHD | | 5 | | | | 2 | | | 1 | | | 180 | |
| | * | 110_READ | | 5 | | | | 3 | | | 1 | | | 1125 | |
| | * | 200_MVMT | | 5 | | | | 2 | | | 1 | | | 180 | |
| | * | 210_READ | | 5 | | | | 3 | | | 1 | | | 1125 | |
| | * | **TOTAL** | | **24** | | | | **12** | | | **6** | | | **2626** | |
| NXD2WQ | * | 000_DRIVER | | 7 | | | | 3 | | | 2 | | | 252 | |
| | * | 100_TC_ONHAND | | 5 | | | | 2 | | | 1 | | | 720 | |
| | * | 110_READ_WRITE_LOOP | | 10 | | | | 4 | | | 2 | | | 12960 | |
| | * | 200_TC_MOVEMENT | | 5 | | | | 2 | | | 1 | | | 720 | |
| | * | 210_READ_WRITE_LOOP | | 10 | | | | 4 | | | 2 | | | 12960 | |
| | * | 300_COMT | | 5 | | | | 2 | | | 1 | | | 720 | |
| | * | 310_COMT_LOOP | | 10 | | | | 4 | | | 2 | | | 12960 | |
| | * | **TOTAL** | | **52** | | | | **21** | | | **11** | | | **41292** | |
| NXD3WQ | * | 000_DRIVER | | 3 | | | | 2 | | | 2 | | | 12 | |
| | * | 100_TC_COMMERCIAL | | 5 | | | | 2 | | | 1 | | | 180 | |
| | * | 110_READ_WRITE_LOOP | | 6 | | | | 3 | | | 1 | | | 1350 | |
| | * | 200_TC_MOVEMENT | | 5 | | | | 2 | | | 1 | | | 180 | |
| | * | 210_READ_WRITE_LOOP | | 6 | | | | 3 | | | 1 | | | 1350 | |
| | * | **TOTAL** | | **25** | | | | **12** | | | **6** | | | **3072** | |
| NXD4WQ | * | 000_DRIVER | | 1 | | | | 1 | | | 1 | | | 0 | |
| | * | 100_TC_MOVEMENT | | 5 | | | | 2 | | | 1 | | | 180 | |
| | * | 110_READ_WRITE_LOOP | | 6 | | | | 3 | | | 1 | | | 1350 | |
| | * | **TOTAL** | | **12** | | | | **6** | | | **3** | | | **1530** | |
| NXD5WQ | * | 000_DRIVER | | 1 | | | | 1 | | | 1 | | | 0 | |
| | * | 100_SUM | | 5 | | | | 2 | | | 1 | | | 180 | |
| | * | 110_READ_WRITE_LOOP | | 6 | | | | 3 | | | 1 | | | 1350 | |
| | * | **TOTAL** | | **12** | | | | **6** | | | **3** | | | **1530** | |
| NXD7WQ | * | 000_DRIVER | | 3 | | | | 2 | | | 1 | | | 27 | |

# Product Measures by Changed Modules

| File Name | * = New Programs | Module/Procedure Name | Pre-Stmts | Post-Stmts | Δ Stmts | Stmts Affected | Pre-McCabe | Post-McCabe | Δ McCabe | Pre-Nest Level | Post-Nest Level | Δ Nest Level | Pre-Info Flow | Post-Info Flow | Δ Info Flow |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | * | 100_HSKP | | 1 | | | | 1 | | | 1 | | | 0 | |
| | * | 200_READ | | 5 | | | | 3 | | | 1 | | | 1125 | |
| | * | 300_WRAPUP | | 3 | | | | 1 | | | 1 | | | 0 | |
| | * | **TOTAL** | | **12** | | | | **7** | | | **4** | | | **1152** | |
| NXIEWQ | * | 000_IE_DRIVER | | 7 | | | | 3 | | | 2 | | | 448 | |
| | * | 100_HSKP | | 38 | | | | 6 | | | 2 | | | 30711638 | |
| | * | 110_SELECT_RECORDS | | 38 | | | | 19 | | | 2 | | | 5429592 | |
| | * | 200_REPORT_BUILDER | | 5 | | | | 4 | | | 2 | | | 1125 | |
| | * | 210_NAF_LOOP | | 5 | | | | 5 | | | 2 | | | 2205 | |
| | * | 211_APOE_LOOP | | 5 | | | | 6 | | | 2 | | | 3645 | |
| | * | 212_APOE_TOTAL | | 11 | | | | 2 | | | 2 | | | 14256 | |
| | * | 2111_APOD_LOOP | | 26 | | | | 10 | | | 2 | | | 2402816 | |
| | * | 2112_APOD_TOTAL | | 20 | | | | 4 | | | 2 | | | 288000 | |
| | * | 220_NAF_TOTAL | | 13 | | | | 4 | | | 2 | | | 15925 | |
| | * | 300_WRAPUP | | 7 | | | | 2 | | | 2 | | | 9072 | |
| | * | 700_HEADERS | | 10 | | | | 7 | | | 2 | | | 12960 | |
| | * | 700_TRAILER | | 5 | | | | 3 | | | 2 | | | 720 | |
| | * | 700_NEW_PAGE | | 7 | | | | 4 | | | 2 | | | 5488 | |
| | * | 700_COMPUTE | | 23 | | | | 2 | | | 2 | | | 7394247 | |
| | * | **TOTAL** | | **220** | | | | **81** | | | **30** | | | **46292137** | |
| YG15WQ | | 1A_STARS_CALL | 41 | 41 | 0 | 1 | 13 | 13 | 0 | 2 | 2 | 0 | 9446400 | 8865225 | -581175 |
| YGABWQ | | 000_DRIVER | 32 | 33 | 1 | 7 | 8 | 10 | 2 | 2 | 2 | 0 | 5120000 | 8449188 | 3329188 |
| | | 100_ABORT_MANIFEST | 56 | 55 | -1 | 1 | 10 | 10 | 0 | 2 | 2 | 0 | 99058400 | 91809520 | -7248880 |
| | | 300_ABORT_DETAIL | 29 | 30 | 1 | 1 | 5 | 6 | 1 | 2 | 2 | 0 | 4827456 | 6768750 | 1941294 |
| | | 1100_BUILD_STATUS_RECORD | 17 | 10 | -7 | 18 | 2 | 1 | -1 | 2 | 1 | -1 | 1332800 | 121000 | -1211800 |
| | | **TOTAL** | **134** | **128** | **-6** | **27** | **25** | **27** | **2** | **8** | **7** | **-1** | **110338656** | **107148458** | **-3190198** |
| YGALWQ | | 000_DRIVER | 69 | 69 | 0 | 2 | 21 | 21 | 0 | 3 | 3 | 0 | 171163125 | 178854900 | 7691775 |
| | | 000_CONTINUE | 5 | 21 | 16 | 16 | 2 | 4 | 2 | 2 | 3 | 1 | 1620 | 979776 | 978156 |
| | | 100_LIFT_U | 73 | 78 | 5 | 5 | 16 | 16 | 0 | 2 | 2 | 0 | 177652800 | 242437182 | 64784382 |
| | | 400_LIFT_DETAIL | 28 | 30 | 2 | 2 | 3 | 3 | 0 | 2 | 2 | 0 | 5976432 | 7620480 | 1644048 |
| | | 860_PLT_BALANCE_CHECK | 30 | 35 | 5 | 11 | 7 | 9 | 2 | 3 | 3 | 0 | 192000 | 283500 | 91500 |
| | | 870_MANFST_BALANCE_CHECK | 28 | 32 | 4 | 6 | 6 | 6 | 0 | 3 | 3 | 0 | 548800 | 720000 | 171200 |
| | | 1110_CUBE_CONVER | 35 | 36 | 1 | 1 | 21 | 22 | 1 | 2 | 2 | 0 | 126000 | 186624 | 60624 |
| | | 1500_DELETE_ASIF_MANIFEST | 18 | 39 | 21 | 23 | 2 | 7 | 5 | 2 | 4 | 2 | 778752 | 11372400 | 10593648 |
| | | **TOTAL** | **286** | **340** | **54** | **66** | **78** | **88** | **10** | **19** | **22** | **3** | **356439529** | **442454862** | **86015333** |
| YGLFWQ | | 000_DRIVER | 51 | 56 | 5 | 5 | 17 | 19 | 2 | 3 | 3 | 0 | 86455404 | 118065024 | 31609620 |
| | | 000_CONTINUE | 21 | 20 | -1 | 1 | 5 | 4 | -1 | 3 | 3 | 0 | 734349 | 699380 | -34969 |
| | | 100_LIFT_U | 68 | 66 | -2 | 4 | 13 | 12 | -1 | 2 | 2 | 0 | 157313988 | 137618976 | -19695012 |
| | | 860_PLT_BALANCE_CHECK | 26 | 33 | 7 | 7 | 6 | 9 | 3 | 3 | 3 | 0 | 109850 | 211200 | 101350 |
| | | 870_MANFST_BALANCE_CHECK | 24 | 30 | 6 | 6 | 5 | 6 | 1 | 2 | 3 | 1 | 345600 | 588000 | 242400 |
| | | 1110_CUBE_CONVERT | 34 | 36 | 2 | 2 | 21 | 22 | 1 | 2 | 2 | 0 | 122400 | 186624 | 64224 |
| | | 1500_BUILD_STATUS_RECORD | 12 | 9 | -3 | 13 | 2 | 1 | -1 | 2 | 1 | -1 | 264268 | 72900 | -191368 |
| | | **TOTAL** | **236** | **250** | **14** | **38** | **69** | **73** | **4** | **17** | **17** | **0** | **245345859** | **257442104** | **12096245** |
| YGOBWQ | | 000_A_SEND_REPORT | 34 | 32 | -2 | 2 | 11 | 11 | 0 | 2 | 2 | 0 | 705024 | 663552 | -41472 |
| | | 030_PLT_BALANCE_CHECK | 31 | 36 | 5 | 10 | 9 | 12 | 3 | 3 | 3 | 0 | 642816 | 1040400 | 397584 |
| | | **TOTAL** | **65** | **68** | **3** | **12** | **20** | **23** | **3** | **5** | **5** | **0** | **1347840** | **1703952** | **356112** |
| YGORWQ | | 515_FORMAT_CALL_RBTCHO | 33 | 32 | -1 | 1 | 7 | 7 | 0 | 2 | 2 | 0 | 1427712 | 1384448 | -43264 |
| | | 520_BD700_SEND_DRIVER | 5 | 3 | -2 | 2 | 3 | 2 | -1 | 2 | 1 | -1 | 320 | 48 | -272 |
| | | **TOTAL** | **38** | **35** | **-3** | **3** | **10** | **9** | **-1** | **4** | **3** | **-1** | **1428032** | **1384496** | **-43536** |
| YGPFWQ | | 1_START_PROCESSING | 43 | 43 | 0 | 1 | 9 | 9 | 0 | 2 | 2 | 0 | 3973888 | 4486147 | 512259 |
| | | 4_DELETE_RECORD | 6 | 17 | 11 | 11 | 2 | 4 | 2 | 2 | 4 | 2 | 3456 | 49572 | 46116 |
| | | **TOTAL** | **49** | **60** | **11** | **12** | **11** | **13** | **2** | **4** | **6** | **2** | **3977344** | **4535719** | **558375** |
| NXMBWQ | | 000_MB_DRIVER | 13 | 15 | 2 | 2 | 7 | 8 | 1 | 2 | 2 | 0 | 4212 | 11760 | 7548 |

# Product Measures by Changed Modules

| File Name | * = New Programs | Module/Procedure Name | Pre-Stmts | Post-Stmts | Δ Stmts | Stmts Affected | Pre-McCabe | Post-McCabe | Δ McCabe | Pre-Nest Level | Post-Nest Level | Δ Nest Level | Pre-Info Flow | Post-Info Flow | Δ Info Flow |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 100_HSKP | 24 | 25 | 1 | 1 | 3 | 4 | 1 | 1 | 2 | 1 | 9255384 | 11390625 | 2135241 |
| | | 110_SELECT_INPUT | 6 | 8 | 2 | 2 | 3 | 5 | 2 | 1 | 2 | 1 | 486 | 1152 | 666 |
| | | 200_BUILD_REPORT | 19 | 22 | 3 | 3 | 6 | 8 | 2 | 2 | 2 | 0 | 112651 | 301158 | 188507 |
| | | 210_TOTALS_TABLE_SEARCH | 6 | 7 | 1 | 1 | 4 | 5 | 1 | 1 | 2 | 1 | 1350 | 2268 | 918 |
| | | 220_UPDATE_TABLE | 60 | 61 | 1 | 1 | 10 | 11 | 1 | 2 | 2 | 0 | 150000 | 184525 | 34525 |
| | | 230_FORMAT_DETAIL_LINE | 23 | 24 | 1 | 1 | 1 | 2 | 1 | 1 | 2 | 1 | 5888828 | 6690816 | 801988 |
| | | 300_WRAP_UP | 9 | 10 | 1 | 1 | 2 | 3 | 1 | 1 | 2 | 1 | 9216 | 16000 | 6784 |
| | | 310_UNLOAD_TABLE | 13 | 14 | 1 | 1 | 2 | 3 | 1 | 1 | 2 | 1 | 29952 | 50400 | 20448 |
| | | 320_PORT_RECAP_PROCESS | 10 | 11 | 1 | 1 | 2 | 3 | 1 | 1 | 2 | 1 | 7290 | 14256 | 6966 |
| | | 321_READ_SORTED_TABLE | 10 | 11 | 1 | 1 | 3 | 4 | 1 | 2 | 2 | 0 | 4000 | 6875 | 2875 |
| | | **TOTAL** | **193** | **208** | **15** | **15** | **43** | **56** | **13** | **15** | **22** | **7** | **15463369** | **18669835** | **3206466** |
| NXIFWQ | * | 000_IF_DRIVER | | 6 | | | | 3 | | | 2 | | | 384 | |
| | * | 100_HOUSEKEEPING | | 32 | | | | 7 | | | 2 | | | 15235200 | |
| | * | 110_SELECT_HISTORY_RECORDS | | 16 | | | | 11 | | | 2 | | | 69696 | |
| | * | 200_REPORT_BUILDER | | 7 | | | | 6 | | | 2 | | | 12348 | |
| | * | 210_CALC_DTL_WEIGHTS_SHIPMENTS | | 3 | | | | 1 | | | 1 | | | 108 | |
| | * | 220_DETAIL_PROCESS | | 11 | | | | 1 | | | 1 | | | 65219 | |
| | * | 230_TOTAL_PROCESS | | 9 | | | | 2 | | | 1 | | | 14400 | |
| | * | 300_WRAPUP | | 10 | | | | 2 | | | 2 | | | 49000 | |
| | * | 700_NEW_PAGE | | 7 | | | | 4 | | | 2 | | | 2800 | |
| | * | 700_HEADERS | | 8 | | | | 5 | | | 2 | | | 6272 | |
| | * | 700_TRAILER | | 5 | | | | 3 | | | 2 | | | 1125 | |
| | * | 700_READ_SORTED_FILE_GFRC | | 2 | | | | 2 | | | 1 | | | 18 | |
| | * | **TOTAL** | | **116** | | | | **47** | | | **20** | | | **15456570** | |
| YGDRWQ | | 2_PROCESS | 45 | 49 | 4 | 4 | 11 | 11 | 0 | 3 | 3 | 0 | 12545280 | 17640000 | 5094720 |
| | | 3_WRAP_UP | 4 | 2 | -2 | 2 | 1 | 1 | 0 | 1 | 1 | 0 | 324 | 18 | -306 |
| | | 20_DGDH_SPAWNED | 15 | 14 | -1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 6000 | 5600 | -400 |
| | | **TOTAL** | **64** | **65** | **1** | **7** | **13** | **13** | **0** | **5** | **5** | **0** | **12551604** | **17645618** | **5094014** |
| YGCTWQ | * | 00000_BEGIN | | 28 | | | | 5 | | | 4 | | 252700 | | |
| | * | 10000_END_OF_DAY_PROCESS | | 14 | | | | 2 | | | 2 | | 89600 | | |
| | * | 20000_READ_SYS_ID_FILE | | 16 | | | | 5 | | | 3 | | 129600 | | |
| | * | 21000_LOAD_TRANS_TABLE | | 5 | | | | 2 | | | 1 | | 6480 | | |
| | * | 21100_LOAD_SYSID_DATA | | 1 | | | | 1 | | | 1 | | 9 | | |
| | * | 30000_FIND_PORT | | 15 | | | | 3 | | | 3 | | 10935 | | |
| | * | 40000_ADD | | 76 | | | | 22 | | | 3 | | 19305216 | | |
| | * | 41000_SEND_MESSAGE | | 28 | | | | 5 | | | 3 | | 444528 | | |
| | * | 42000_CHANGE_SYSTEM_STATUS | | 19 | | | | 6 | | | 5 | | 75411 | | |
| | * | 42100_SET_SYSTEM_STATUS | | 15 | | | | 3 | | | 3 | | 18375 | | |
| | * | 42110_SET_CHANGE_DATE | | 4 | | | | 1 | | | 1 | | 4096 | | |
| | * | 42120_CALL_15 | | 5 | | | | 2 | | | 2 | | 980 | | |
| | * | 50000_STORE_TO_SYS_ID | | 8 | | | | 3 | | | 1 | | 7200 | | |
| | * | 51000_READ_SYS_ID_FILE_AGAIN | | 14 | | | | 8 | | | 3 | | 50400 | | |
| | * | 51100_SEARCH_TRANS_TABLE | | 3 | | | | 2 | | | 2 | | 192 | | |
| | * | 51110_STORE_DATA | | 8 | | | | 3 | | | 2 | | 7200 | | |
| | * | 51111_STORE_TRANS_INFO | | 1 | | | | 1 | | | 1 | | 16 | | |
| | * | 51200_REPLACE_RECORD | | 9 | | | | 2 | | | 2 | | 3600 | | |
| | * | 60000_PROCESS_TRANS_RPT | | 10 | | | | 4 | | | 1 | | 39690 | | |
| | * | 61000_BUILD_AND_SEND | | 29 | | | | 7 | | | 3 | | 1841616 | | |
| | * | 61100_SET_HEADERS | | 4 | | | | 2 | | | 2 | | 400 | | |
| | * | 61200_BUILD_DRIVER | | 5 | | | | 2 | | | 1 | | 8820 | | |
| | * | 61210_ACCESS_2ND_DEMENSION | | 1 | | | | 1 | | | 1 | | 16 | | |
| | * | 61300_SEND_TRANS_RPT | | 22 | | | | 3 | | | 3 | | 130438 | | |
| | * | 61400_SEARCH_PORT_TABLE | | 6 | | | | 3 | | | 3 | | 384 | | |

# Product Measures by Changed Modules

| File Name | * = New Programs | Module/Procedure Name | Pre-Stmts | Post-Stmts | Δ Stmts | Stmts Affected | Pre-McCabe | Post-McCabe | Δ McCabe | Pre-Nest Level | Post-Nest Level | Δ Nest Level | Pre-Info Flow | Post-Info Flow | Δ Info Flow |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | * | 70000_RZLT_CHECK | | 9 | | | | 2 | | | 2 | | 3600 | | |
| | * | 80000_FALL_THRU | | 1 | | | | 1 | | | 1 | | 0 | | |
| | * | **TOTAL** | | **356** | | | | **101** | | | **59** | | **22431502** | | |
| YGUCWQ | | 320_I_CALL | 159 | 163 | 4 | 4 | 42 | 43 | 1 | 5 | 5 | 0 | 1887015975 | 2070441648 | 183425673 |
| | | 1400_CONTROL_RECORD | 13 | 15 | 2 | 2 | 3 | 3 | 0 | 3 | 3 | 0 | 67392 | 181500 | 114108 |
| | | **TOTAL** | **172** | **178** | **6** | **6** | **45** | **46** | **1** | **8** | **8** | **0** | **1887083367** | **2070623148** | **183539781** |
| YGDGWQ | | 300_ADD | 6 | 6 | 0 | 2 | 2 | 2 | 0 | 2 | 2 | 0 | 96 | 96 | 0 |
| YGRNWQ | | 000_START | 33 | 41 | 8 | 8 | 7 | 10 | 3 | 4 | 5 | 1 | 1597200 | 4628736 | 3031536 |
| | | 100_BUILD_ADVANCE | 33 | 40 | 7 | 7 | 8 | 11 | 3 | 2 | 2 | 0 | 5765892 | 11924640 | 6158748 |
| | | 100_ADD_ADVANCE | 12 | 22 | 10 | 10 | 4 | 8 | 4 | 3 | 3 | 0 | 10800 | 95832 | 85032 |
| | | 200_NO_HIT | 38 | 58 | 20 | 20 | 11 | 18 | 7 | 3 | 3 | 0 | 3558168 | 12272800 | 8714632 |
| | | 300_UPDATE_ADVANCE | 14 | 17 | 3 | 3 | 3 | 4 | 1 | 2 | 2 | 0 | 201600 | 347633 | 146033 |
| | | 910_PROJ_CODE_CHK | | 3 | | | | 3 | | | 2 | | | 12 | |
| | | 920_CODE_J_CHK | | 4 | | | | 6 | | | 2 | | | 784 | |
| | | **TOTAL** | **130** | **185** | **48** | **48** | **33** | **60** | **18** | **14** | **19** | **1** | **11133660** | **29270437** | **18135981** |
| YGADWQ | | 000_INITIALIZE | 1 | 4 | 3 | 3 | 1 | 2 | 1 | 1 | 2 | 1 | 0 | 16 | 16 |
| | | 010_DRIVER_PARA | 23 | 34 | 11 | 11 | 20 | 24 | 4 | 5 | 5 | 0 | 178112 | 827424 | 649312 |
| | | 105_GET_PRIME | 32 | 44 | 12 | 12 | 5 | 8 | 3 | 2 | 3 | 1 | 3276800 | 6353600 | 3076800 |
| | | 200_ADD_DETAIL | 13 | 21 | 8 | 12 | 1 | 5 | 4 | 1 | 2 | 1 | 206388 | 873936 | 667548 |
| | | 210_ADD | 23 | 27 | 4 | 4 | 4 | 6 | 2 | 2 | 3 | 1 | 517500 | 1190700 | 673200 |
| | | 310_GET_DETAIL | 27 | 39 | 12 | 12 | 4 | 7 | 3 | 2 | 3 | 1 | 395307 | 797511 | 402204 |
| | | **TOTAL** | **119** | **169** | **50** | **54** | **35** | **52** | **17** | **13** | **18** | **5** | **4574107** | **10043187** | **5469080** |
| YGEDWQ | | 120_EXIT | 19 | 19 | 0 | 1 | 22 | 21 | -1 | 3 | 3 | 0 | 17100 | 17100 | 0 |
| | | 220_EDIT_PRI | 26 | 30 | 4 | 18 | 57 | 52 | -5 | 6 | 2 | -4 | 14976 | 9720 | -5256 |
| | | 300_LIST_ERRORS | 6 | 12 | 6 | 6 | 2 | 4 | 2 | 2 | 3 | 1 | 486 | 5292 | 4806 |
| | | 301_DEFAULT_MSG | 16 | 24 | 8 | 8 | 2 | 4 | 2 | 2 | 2 | 0 | 230400 | 470400 | 240000 |
| | | 303_WRITE | 7 | 15 | 8 | 8 | 1 | 3 | 2 | 1 | 2 | 1 | 1008 | 8640 | 7632 |
| | | 305_MOVES | 60 | 76 | 16 | 16 | 9 | 13 | 4 | 3 | 3 | 0 | 3810240 | 6569136 | 2758896 |
| | | 900_FILL_TABLE | 10 | 16 | 6 | 6 | 2 | 4 | 2 | 2 | 4 | 2 | 4199040 | 7683984 | 3484944 |
| | | 1100_VALID_COUNTERS | 82 | 83 | 1 | 1 | 28 | 29 | 1 | 3 | 3 | 0 | 1845000 | 2689200 | 844200 |
| | | **TOTAL** | **226** | **275** | **49** | **64** | **123** | **130** | **7** | **22** | **22** | **0** | **10118250** | **17453472** | **7335222** |
| YGBTWQ | | 003_COUNTERS | 45 | 46 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| YGOTWQ | | 100_SEND | 24 | 25 | 1 | 1 | 25 | 25 | 0 | 6 | 6 | 0 | 7776 | 14400 | 6624 |
| YGCTWQ | | 00000_BEGIN | 28 | 39 | 11 | 13 | 5 | 6 | 1 | 4 | 5 | 1 | 252700 | 351975 | 99275 |
| | | 40000_ADD | 76 | 68 | -8 | 12 | 22 | 18 | -4 | 3 | 2 | -1 | 19305216 | 7953552 | -11351664 |
| | | 61100_SET_HEADERS | 4 | 11 | 7 | 7 | 2 | 2 | 0 | 2 | 2 | 0 | 400 | 107811 | 107411 |
| | | **TOTAL** | **108** | **118** | **10** | **32** | **29** | **26** | **-3** | **9** | **9** | **0** | **19558316** | **8413338** | **-11144978** |
| NXM1WQ | | 000_DRIVER | 5 | 7 | 2 | 2 | 2 | 3 | 1 | 1 | 2 | 1 | 80 | 700 | 620 |
| | | 100_HOUSEKEEPING | 6 | 7 | 1 | 1 | 2 | 3 | 1 | 1 | 2 | 1 | 3456 | 5488 | 2032 |
| | | 110_LOAD_ADAM_III_CONUS | 9 | 10 | 1 | 1 | 3 | 4 | 1 | 2 | 2 | 0 | 2025 | 3240 | 1215 |
| | | 200_PROCESS_LB | 7 | 8 | 1 | 1 | 8 | 9 | 1 | 2 | 2 | 0 | 4375 | 7200 | 2825 |
| | | 300_AKF_SETTER | 8 | 9 | 1 | 1 | 3 | 4 | 1 | 1 | 2 | 1 | 648 | 1296 | 648 |
| | | 400_WRAPUP | 2 | 3 | 1 | 1 | 1 | 2 | 1 | 1 | 2 | 1 | 0 | 0 | 0 |
| | | 700_READ_LB_MOVEMENT | 4 | 5 | 1 | 1 | 2 | 3 | 1 | 1 | 2 | 1 | 36 | 80 | 44 |
| | | 700_LB_READ_WRITE | 5 | 8 | 3 | 3 | 3 | 5 | 2 | 1 | 2 | 1 | 180 | 800 | 620 |
| | | 700_CHECK_ADAM_III_CONUS | 5 | 6 | 1 | 1 | 3 | 4 | 1 | 1 | 2 | 1 | 500 | 864 | 364 |
| | | **TOTAL** | **51** | **63** | **12** | **12** | **27** | **37** | **10** | **11** | **18** | **7** | **11300** | **19668** | **8368** |
| NXMPWQ | | 000_MP_DRIVER | 7 | 9 | 2 | 2 | 2 | 3 | 1 | 2 | 2 | 0 | 112 | 900 | 788 |
| | | 100_HSKP | 46 | 48 | 2 | 2 | 9 | 11 | 2 | 1 | 2 | 1 | 29292984 | 32039472 | 2746488 |
| | | 110_PZ4_ORIGINATING | 26 | 27 | 1 | 1 | 16 | 17 | 1 | 2 | 2 | 0 | 531674 | 640332 | 108658 |
| | | 120_PZ4_TERMINATING | 26 | 27 | 1 | 1 | 14 | 15 | 1 | 2 | 2 | 0 | 380666 | 470448 | 89782 |
| | | 120_CONTINUE | 24 | 27 | 3 | 3 | 8 | 10 | 2 | 2 | 2 | 0 | 2217934 | 3158028 | 940044 |
| | | 130_PZ3_OLD_MONTH | 38 | 41 | 3 | 3 | 21 | 23 | 2 | 3 | 3 | 0 | 2318342 | 2771600 | 453258 |

# Product Measures by Changed Modules

| File Name | * = New Programs | Module/Procedure Name | Pre-Stmts | Post-Stmts | Δ Stmts | Stmts Affected | Pre-McCabe | Post-McCabe | Δ McCabe | Pre-Nest Level | Post-Nest Level | Δ Nest Level | Pre-Info Flow | Post-Info Flow | Δ Info Flow |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 200_CREATE_FILE_DRIVER | 5 | 6 | 1 | 1 | 2 | 3 | 1 | 1 | 2 | 1 | 1620 | 2646 | 1026 |
| | | 210_CREATE_MINI_FILE | 18 | 19 | 1 | 1 | 17 | 18 | 1 | 4 | 4 | 0 | 127008 | 157339 | 30331 |
| | | 211_ORIG_FOR_MINI_REC | 31 | 34 | 3 | 3 | 24 | 27 | 3 | 2 | 2 | 0 | 1290096 | 1586304 | 296208 |
| | | 212_COMPUTE_PHTS | 54 | 55 | 1 | 1 | 7 | 8 | 1 | 2 | 2 | 0 | 15283296 | 17248000 | 1964704 |
| | | 300_WRAPUP | 45 | 46 | 1 | 1 | 1 | 2 | 1 | 1 | 2 | 1 | 0 | 0 | 0 |
| | | 700_READ | 6 | 7 | 1 | 1 | 2 | 3 | 1 | 1 | 2 | 1 | 216 | 448 | 232 |
| | | 700_ADAM_III_CHECK | 5 | 6 | 1 | 1 | 3 | 4 | 1 | 1 | 2 | 1 | 980 | 1536 | 556 |
| | | 700_COMPUTE_HOURS | 16 | 17 | 1 | 1 | 6 | 7 | 1 | 2 | 2 | 0 | 186624 | 220932 | 34308 |
| | | 700_WRITE_OVER_10_DAY | 8 | 9 | 1 | 1 | 2 | 3 | 1 | 1 | 2 | 1 | 14112 | 21609 | 7497 |
| | | 700_READ_MOVE | 13 | 15 | 2 | 2 | 13 | 15 | 2 | 7 | 9 | 2 | 25168 | 34560 | 9392 |
| | | 700_ERROR_RTN | 4 | 5 | 1 | 1 | 1 | 2 | 1 | 1 | 2 | 1 | 196 | 320 | 124 |
| | | **TOTAL** | **372** | **398** | **26** | **26** | **148** | **171** | **23** | **35** | **44** | **9** | **51671078** | **58354474** | **6683396** |
| NXRPWQ | | 000_DRIVER | 8 | 11 | 3 | 3 | 3 | 5 | 2 | 2 | 2 | 0 | 4608 | 13475 | 8867 |
| | | 100_HSKP | 25 | 27 | 2 | 2 | 7 | 9 | 2 | 2 | 2 | 0 | 608400 | 762048 | 153648 |
| | | 110_VALIDATE_REPORT_CODE | 14 | 16 | 2 | 2 | 6 | 8 | 2 | 2 | 2 | 0 | 14336 | 20736 | 6400 |
| | | 130_INITIALIZE_REPORT_FILE | 18 | 19 | 1 | 1 | 6 | 7 | 1 | 2 | 2 | 0 | 16200 | 23275 | 7075 |
| | | 131_ADDFIL_CALL | 13 | 14 | 1 | 1 | 8 | 9 | 1 | 2 | 2 | 0 | 83200 | 101150 | 17950 |
| | | 200_PROCESS_REPORT_FILE | 7 | 8 | 1 | 1 | 4 | 5 | 1 | 1 | 2 | 1 | 2800 | 4608 | 1808 |
| | | 200_SEARCH_AT_END | 3 | 4 | 1 | 1 | 4 | 5 | 1 | 1 | 2 | 1 | 300 | 576 | 276 |
| | | 200_SEARCH_WHEN_1 | 2 | 3 | 1 | 1 | 2 | 3 | 1 | 1 | 2 | 1 | 8 | 27 | 19 |
| | | 200_SEARCH_WHEN_2 | 4 | 5 | 1 | 1 | 4 | 5 | 1 | 1 | 2 | 1 | 900 | 1620 | 720 |
| | | 210_WRITE_AND_READ_NEXT | 7 | 9 | 2 | 2 | 3 | 5 | 2 | 2 | 2 | 0 | 2268 | 3969 | 1701 |
| | | 220_SKIP_THIS_STATION | 4 | 6 | 2 | 2 | 2 | 4 | 2 | 1 | 2 | 1 | 36 | 96 | 60 |
| | | 300_CLOSE | 11 | 13 | 2 | 2 | 5 | 7 | 2 | 2 | 2 | 0 | 70400 | 93925 | 23525 |
| | | 310_DISPLAY | 8 | 9 | 1 | 1 | 2 | 3 | 1 | 2 | 2 | 0 | 512 | 900 | 388 |
| | | 700_ADDFIL_ERROR_CHECK | 11 | 12 | 1 | 1 | 3 | 4 | 1 | 2 | 2 | 0 | 33275 | 43200 | 9925 |
| | | 700_READ_REQUEST_FILE | 4 | 6 | 2 | 2 | 2 | 4 | 2 | 1 | 2 | 1 | 64 | 150 | 86 |
| | | **TOTAL** | **139** | **162** | **23** | **23** | **61** | **83** | **22** | **24** | **30** | **6** | **837307** | **1069755** | **232448** |

*Appendix F*

# Changes in Raw Product Measures by Module

| File | Module/Procedure | Δ Stmts | Affected Stmts | Δ McCabe | Δ Nest Level | Δ Info Flow |
|---|---|---|---|---|---|---|
| NXR7WQ | 000_CLOSE | 1 | 1 | 0 | 0 | 2268 |
| | 500_TCN_CHECK | 1 | 1 | 0 | 0 | 1696 |
| NXICWQ | 100_HSKP | 1 | 1 | 0 | 0 | 3962875 |
| | 700_SELECT_AND_MOVE | -2 | 4 | -1 | 0 | -62280 |
| NXIDWQ | 110_SELECT | -4 | 4 | 0 | 0 | -1085888 |
| | 720_PROCESS | 0 | 1 | -1 | 0 | -720896 |
| NXMHWQ | 000_CONTROL | 1 | 2 | 0 | 0 | 8400 |
| | 131_SELECT_INPUT | 3 | 3 | 1 | 0 | 255180 |
| | 420_AIR_TOTALS | 25 | 25 | 4 | 0 | 387377329 |
| | 440_TRK_TOTALS | 23 | 23 | 2 | 0 | 118977174 |
| NXPMWQ | 211_ALLOCATE_YESTERDAY_FILE | 6 | 6 | 3 | 0 | 35955 |
| NXRGWQ | 220_PROCESS | 5 | 7 | 3 | 0 | 1050885 |
| NXPYWQ | 200_REFORMAT_INPUT_FILE | 3 | 6 | 2 | 0 | 73872 |
| NXRPWQ | 131_ADDFIL_CALL | 5 | 7 | 3 | 0 | 59000 |
| NXU1WQ | 210_ALLOCATE_AKF_FILE | 10 | 10 | 3 | 0 | 1025936 |
| NXD0WQ | 000_DRIVER | 3 | 3 | 2 | 1 | 27 |
| | 100_HSKP | 1 | 1 | 1 | 1 | 0 |
| | 200_READ | 5 | 5 | 3 | 1 | 1125 |
| | 300_WRAPUP | 3 | 3 | 1 | 1 | 0 |
| NXD1WQ | 000_DRIVER | 4 | 4 | 2 | 2 | 16 |
| | 100_ONHD | 5 | 5 | 2 | 1 | 180 |
| | 110_READ | 5 | 5 | 3 | 1 | 1125 |
| | 200_MVMT | 5 | 5 | 2 | 1 | 180 |
| | 210_READ | 5 | 5 | 3 | 1 | 1125 |
| NXD2WQ | 000_DRIVER | 7 | 7 | 3 | 2 | 252 |
| | 100_TC_ONHAND | 5 | 5 | 2 | 1 | 720 |
| | 110_READ_WRITE_LOOP | 10 | 10 | 4 | 2 | 12960 |
| | 200_TC_MOVEMENT | 5 | 5 | 2 | 1 | 720 |
| | 210_READ_WRITE_LOOP | 10 | 10 | 4 | 2 | 12960 |
| | 300_COMT | 5 | 5 | 2 | 1 | 720 |
| | 310_COMT_LOOP | 10 | 10 | 4 | 2 | 12960 |
| NXD3WQ | 000_DRIVER | 3 | 3 | 2 | 2 | 12 |
| | 100_TC_COMMERCIAL | 5 | 5 | 2 | 1 | 180 |
| | 110_READ_WRITE_LOOP | 6 | 6 | 3 | 1 | 1350 |
| | 200_TC_MOVEMENT | 5 | 5 | 2 | 1 | 180 |
| | 210_READ_WRITE_LOOP | 6 | 6 | 3 | 1 | 1350 |
| NXD4WQ | 000_DRIVER | 1 | 1 | 1 | 1 | 0 |
| | 100_TC_MOVEMENT | 5 | 5 | 2 | 1 | 180 |
| | 110_READ_WRITE_LOOP | 6 | 6 | 3 | 1 | 1350 |
| NXD5WQ | 000_DRIVER | 1 | 1 | 1 | 1 | 0 |
| | 100_SUM | 5 | 5 | 2 | 1 | 180 |

## Changes in Raw Product Measures by Module

| File | Module/Procedure | Δ Stmts | Affected Stmts | Δ McCabe | Δ Nest Level | Δ Info Flow |
|---|---|---|---|---|---|---|
| | 110_READ_WRITE_LOOP | 6 | 6 | 3 | 1 | 1350 |
| NXD7WQ | 000_DRIVER | 3 | 3 | 2 | 1 | 27 |
| | 100_HSKP | 1 | 1 | 1 | 1 | 0 |
| | 200_READ | 5 | 5 | 3 | 1 | 1125 |
| | 300_WRAPUP | 3 | 3 | 1 | 1 | 0 |
| NXIEWQ | 000_IE_DRIVER | 7 | 7 | 3 | 2 | 448 |
| | 100_HSKP | 38 | 38 | 6 | 2 | 30711638 |
| | 110_SELECT_RECORDS | 38 | 38 | 19 | 2 | 5429592 |
| | 200_REPORT_BUILDER | 5 | 5 | 4 | 2 | 1125 |
| | 210_NAF_LOOP | 5 | 5 | 5 | 2 | 2205 |
| | 211_APOE_LOOP | 5 | 5 | 6 | 2 | 3645 |
| | 212_APOE_TOTAL | 11 | 11 | 2 | 2 | 14256 |
| | 2111_APOD_LOOP | 26 | 26 | 10 | 2 | 2402816 |
| | 2112_APOD_TOTAL | 20 | 20 | 4 | 2 | 288000 |
| | 220_NAF_TOTAL | 13 | 13 | 4 | 2 | 15925 |
| | 300_WRAPUP | 7 | 7 | 2 | 2 | 9072 |
| | 700_HEADERS | 10 | 10 | 7 | 2 | 12960 |
| | 700_TRAILER | 5 | 5 | 3 | 2 | 720 |
| | 700_NEW_PAGE | 7 | 7 | 4 | 2 | 5488 |
| | 700_COMPUTE | 23 | 23 | 2 | 2 | 7394247 |
| YG15WQ | 1A_STARS_CALL | 0 | 1 | 0 | 0 | -581175 |
| YGABWQ | 000_DRIVER | 1 | 7 | 2 | 0 | 3329188 |
| | 100_ABORT_MANIFEST | -1 | 1 | 0 | 0 | -7248880 |
| | 300_ABORT_DETAIL | 1 | 1 | 1 | 0 | 1941294 |
| | 1100_BUILD_STATUS_RECORD | -7 | 18 | -1 | -1 | -1211800 |
| YGALWQ | 000_DRIVER | 0 | 2 | 0 | 0 | 7691775 |
| | 000_CONTINUE | 16 | 16 | 2 | 1 | 978156 |
| | 100_LIFT_U | 5 | 5 | 0 | 0 | 64784382 |
| | 400_LIFT_DETAIL | 2 | 2 | 0 | 0 | 1644048 |
| | 860_PLT_BALANCE_CHECK | 5 | 11 | 2 | 0 | 91500 |
| | 870_MANFST_BALANCE_CHECK | 4 | 6 | 0 | 0 | 171200 |
| | 1110_CUBE_CONVER | 1 | 1 | 1 | 0 | 60624 |
| | 1500_DELETE_ASIF_MANIFEST | 21 | 23 | 5 | 2 | 10593648 |
| YGLFWQ | 000_DRIVER | 5 | 5 | 2 | 0 | 31609620 |
| | 000_CONTINUE | -1 | 1 | -1 | 0 | -34969 |
| | 100_LIFT_U | -2 | 4 | -1 | 0 | -19695012 |
| | 860_PLT_BALANCE_CHECK | 7 | 7 | 3 | 0 | 101350 |
| | 870_MANFST_BALANCE_CHECK | 6 | 6 | 1 | 1 | 242400 |
| | 1110_CUBE_CONVERT | 2 | 2 | 1 | 0 | 64224 |
| | 1500_BUILD_STATUS_RECORD | -3 | 13 | -1 | -1 | -191368 |
| YGOBWQ | 000_A_SEND_REPORT | -2 | 2 | 0 | 0 | -41472 |

## Changes in Raw Product Measures by Module

| File | Module/Procedure | Δ Stmts | Affected Stmts | Δ McCabe | Δ Nest Level | Δ Info Flow |
|---|---|---|---|---|---|---|
| | 030_PLT_BALANCE_CHECK | 5 | 10 | 3 | 0 | 397584 |
| YGORWQ | 515_FORMAT_CALL_RBTCHO | -1 | 1 | 0 | 0 | -43264 |
| | 520_BD700_SEND_DRIVER | -2 | 2 | -1 | -1 | -272 |
| YGPFWQ | 1_START_PROCESSING | 0 | 1 | 0 | 0 | 512259 |
| | 4_DELETE_RECORD | 11 | 11 | 2 | 2 | 46116 |
| NXMBWQ | 000_MB_DRIVER | 2 | 2 | 1 | 0 | 7548 |
| | 100_HSKP | 1 | 1 | 1 | 1 | 2135241 |
| | 110_SELECT_INPUT | 2 | 2 | 2 | 1 | 666 |
| | 200_BUILD_REPORT | 3 | 3 | 2 | 0 | 188507 |
| | 210_TOTALS_TABLE_SEARCH | 1 | 1 | 1 | 1 | 918 |
| | 220_UPDATE_TABLE | 1 | 1 | 1 | 0 | 34525 |
| | 230_FORMAT_DETAIL_LINE | 1 | 1 | 1 | 1 | 801988 |
| | 300_WRAP_UP | 1 | 1 | 1 | 1 | 6784 |
| | 310_UNLOAD_TABLE | 1 | 1 | 1 | 1 | 20448 |
| | 320_PORT_RECAP_PROCESS | 1 | 1 | 1 | 1 | 6966 |
| | 321_READ_SORTED_TABLE | 1 | 1 | 1 | 0 | 2875 |
| NXIFWQ | 000_IF_DRIVER | 6 | 6 | 3 | 2 | 384 |
| | 100_HOUSEKEEPING | 32 | 32 | 7 | 2 | 15235200 |
| | 110_SELECT_HISTORY_RECORDS | 16 | 16 | 11 | 2 | 69696 |
| | 200_REPORT_BUILDER | 7 | 7 | 6 | 2 | 12348 |
| | 210_CALC_DTL_WEIGHTS_SHIPMENTS | 3 | 3 | 1 | 1 | 108 |
| | 220_DETAIL_PROCESS | 11 | 11 | 1 | 1 | 65219 |
| | 230_TOTAL_PROCESS | 9 | 9 | 2 | 1 | 14400 |
| | 300_WRAPUP | 10 | 10 | 2 | 2 | 49000 |
| | 700_NEW_PAGE | 7 | 7 | 4 | 2 | 2800 |
| | 700_HEADERS | 8 | 8 | 5 | 2 | 6272 |
| | 700_TRAILER | 5 | 5 | 3 | 2 | 1125 |
| | 700_READ_SORTED_FILE_GFRC | 2 | 2 | 2 | 1 | 18 |
| YGDRWQ | 2_PROCESS | 4 | 4 | 0 | 0 | 5094720 |
| | 3_WRAP_UP | -2 | 2 | 0 | 0 | -306 |
| | 20_DGDH_SPAWNED | -1 | 1 | 0 | 0 | -400 |
| YGCTWQ | 00000_BEGIN | 28 | 28 | 5 | 4 | 252700 |
| | 10000_END_OF_DAY_PROCESS | 14 | 14 | 2 | 2 | 89600 |
| | 20000_READ_SYS_ID_FILE | 16 | 16 | 5 | 3 | 129600 |
| | 21000_LOAD_TRANS_TABLE | 5 | 5 | 2 | 1 | 6480 |
| | 21100_LOAD_SYSID_DATA | 1 | 1 | 1 | 1 | 9 |
| | 30000_FIND_PORT | 15 | 15 | 3 | 3 | 10935 |
| | 40000_ADD | 76 | 76 | 22 | 3 | 19305216 |
| | 41000_SEND_MESSAGE | 28 | 28 | 5 | 3 | 444528 |
| | 42000_CHANGE_SYSTEM_STATUS | 19 | 19 | 6 | 5 | 75411 |
| | 42100_SET_SYSTEM_STATUS | 15 | 15 | 3 | 3 | 18375 |

## Changes in Raw Product Measures by Module

| File | Module/Procedure | Δ Stmts | Affected Stmts | Δ McCabe | Δ Nest Level | Δ Info Flow |
|---|---|---|---|---|---|---|
| | 42110_SET_CHANGE_DATE | 4 | 4 | 1 | 1 | 4096 |
| | 42120_CALL_15 | 5 | 5 | 2 | 2 | 980 |
| | 50000_STORE_TO_SYS_ID | 8 | 8 | 3 | 1 | 7200 |
| | 51000_READ_SYS_ID_FILE_AGAIN | 14 | 14 | 8 | 3 | 50400 |
| | 51100_SEARCH_TRANS_TABLE | 3 | 3 | 2 | 2 | 192 |
| | 51110_STORE_DATA | 8 | 8 | 3 | 2 | 7200 |
| | 51111_STORE_TRANS_INFO | 1 | 1 | 1 | 1 | 16 |
| | 51200_REPLACE_RECORD | 9 | 9 | 2 | 2 | 3600 |
| | 60000_PROCESS_TRANS_RPT | 10 | 10 | 4 | 1 | 39690 |
| | 61000_BUILD_AND_SEND | 29 | 29 | 7 | 3 | 1841616 |
| | 61100_SET_HEADERS | 4 | 4 | 2 | 2 | 400 |
| | 61200_BUILD_DRIVER | 5 | 5 | 2 | 1 | 8820 |
| | 61210_ACCESS_2ND_DEMENSION | 1 | 1 | 1 | 1 | 16 |
| | 61300_SEND_TRANS_RPT | 22 | 22 | 3 | 3 | 130438 |
| | 61400_SEARCH_PORT_TABLE | 6 | 6 | 3 | 3 | 384 |
| | 70000_RZLT_CHECK | 9 | 9 | 2 | 2 | 3600 |
| | 80000_FALL_THRU | 1 | 1 | 1 | 1 | 0 |
| YGUCWQ | 320_I_CALL | 4 | 4 | 1 | 0 | 183425673 |
| | 1400_CONTROL_RECORD | 2 | 2 | 0 | 0 | 114108 |
| YGDGWQ | 300_ADD | 0 | 2 | 0 | 0 | 0 |
| YGRNWQ | 000_START | 8 | 8 | 3 | 1 | 3031536 |
| | 100_BUILD_ADVANCE | 7 | 7 | 3 | 0 | 6158748 |
| | 100_ADD_ADVANCE | 10 | 10 | 4 | 0 | 85032 |
| | 200_NO_HIT | 20 | 20 | 7 | 0 | 8714632 |
| | 300_UPDATE_ADVANCE | 3 | 3 | 1 | 0 | 146033 |
| | 910_PROJ_CODE_CHK | 3 | 3 | 3 | 2 | 12 |
| | 920_CODE_J_CHK | 4 | 4 | 6 | 2 | 784 |
| YGADWQ | 000_INITIALIZE | 3 | 3 | 1 | 1 | 16 |
| | 010_DRIVER_PARA | 11 | 11 | 4 | 0 | 649312 |
| | 105_GET_PRIME | 12 | 12 | 3 | 1 | 3076800 |
| | 200_ADD_DETAIL | 8 | 12 | 4 | 1 | 667548 |
| | 210_ADD | 4 | 4 | 2 | 1 | 673200 |
| | 310_GET_DETAIL | 12 | 12 | 3 | 1 | 402204 |
| YGEDWQ | 120_EXIT | 0 | 1 | -1 | 0 | 0 |
| | 220_EDIT_PRI | 4 | 18 | -5 | -4 | -5256 |
| | 300_LIST_ERRORS | 6 | 6 | 2 | 1 | 4806 |
| | 301_DEFAULT_MSG | 8 | 8 | 2 | 0 | 240000 |
| | 303_WRITE | 8 | 8 | 2 | 1 | 7632 |
| | 305_MOVES | 16 | 16 | 4 | 0 | 2758896 |
| | 900_FILL_TABLE | 6 | 6 | 2 | 2 | 3484944 |
| | 1100_VALID_COUNTERS | 1 | 1 | 1 | 0 | 844200 |

# Changes in Raw Product Measures by Module

| File | Module/Procedure | Δ Stmts | Affected Stmts | Δ McCabe | Δ Nest Level | Δ Info Flow |
|---|---|---|---|---|---|---|
| YGBTWQ | 003_COUNTERS | 1 | 1 | 0 | 0 | 0 |
| YGOTWQ | 100_SEND | 1 | 1 | 0 | 0 | 6624 |
| YGCTWQ | 00000_BEGIN | 11 | 13 | 1 | 1 | 99275 |
| | 40000_ADD | -8 | 12 | -4 | -1 | -11351664 |
| | 61100_SET_HEADERS | 7 | 7 | 0 | 0 | 107411 |
| NXM1WQ | 000_DRIVER | 2 | 2 | 1 | 1 | 620 |
| | 100_HOUSEKEEPING | 1 | 1 | 1 | 1 | 2032 |
| | 110_LOAD_ADAM_III_CONUS | 1 | 1 | 1 | 0 | 1215 |
| | 200_PROCESS_LB | 1 | 1 | 1 | 0 | 2825 |
| | 300_AKF_SETTER | 1 | 1 | 1 | 1 | 648 |
| | 400_WRAPUP | 1 | 1 | 1 | 1 | 0 |
| | 700_READ_LB_MOVEMENT | 1 | 1 | 1 | 1 | 44 |
| | 700_LB_READ_WRITE | 3 | 3 | 2 | 1 | 620 |
| | 700_CHECK_ADAM_III_CONUS | 1 | 1 | 1 | 1 | 364 |
| NXMPWQ | 000_MP_DRIVER | 2 | 2 | 1 | 0 | 788 |
| | 100_HSKP | 2 | 2 | 2 | 1 | 2746488 |
| | 110_PZ4_ORIGINATING | 1 | 1 | 1 | 0 | 108658 |
| | 120_PZ4_TERMINATING | 1 | 1 | 1 | 0 | 89782 |
| | 120_CONTINUE | 3 | 3 | 2 | 0 | 940044 |
| | 130_PZ3_OLD_MONTH | 3 | 3 | 2 | 0 | 453258 |
| | 200_CREATE_FILE_DRIVER | 1 | 1 | 1 | 1 | 1026 |
| | 210_CREATE_MINI_FILE | 1 | 1 | 1 | 0 | 30331 |
| | 211_ORIG_FOR_MINI_REC | 3 | 3 | 3 | 0 | 296208 |
| | 212_COMPUTE_PHTS | 1 | 1 | 1 | 0 | 1964704 |
| | 300_WRAPUP | 1 | 1 | 1 | 1 | 0 |
| | 700_READ | 1 | 1 | 1 | 1 | 232 |
| | 700_ADAM_III_CHECK | 1 | 1 | 1 | 1 | 556 |
| | 700_COMPUTE_HOURS | 1 | 1 | 1 | 0 | 34308 |
| | 700_WRITE_OVER_10_DAY | 1 | 1 | 1 | 1 | 7497 |
| | 700_READ_MOVE | 2 | 2 | 2 | 2 | 9392 |
| | 700_ERROR_RTN | 1 | 1 | 1 | 1 | 124 |
| NXRPWQ | 000_DRIVER | 3 | 3 | 2 | 0 | 8867 |
| | 100_HSKP | 2 | 2 | 2 | 0 | 153648 |
| | 110_VALIDATE_REPORT_CODE | 2 | 2 | 2 | 0 | 6400 |
| | 130_INITIALIZE_REPORT_FILE | 1 | 1 | 1 | 0 | 7075 |
| | 131_ADDFIL_CALL | 1 | 1 | 1 | 0 | 17950 |
| | 200_PROCESS_REPORT_FILE | 1 | 1 | 1 | 1 | 1808 |
| | 200_SEARCH_AT_END | 1 | 1 | 1 | 1 | 276 |
| | 200_SEARCH_WHEN_1 | 1 | 1 | 1 | 1 | 19 |
| | 200_SEARCH_WHEN_2 | 1 | 1 | 1 | 1 | 720 |
| | 210_WRITE_AND_READ_NEXT | 2 | 2 | 2 | 0 | 1701 |

## Changes in Raw Product Measures by Module

| File | Module/Procedure | Δ Stmts | Affected Stmts | Δ McCabe | Δ Nest Level | Δ Info Flow |
|---|---|---|---|---|---|---|
| | 220_SKIP_THIS_STATION | 2 | 2 | 2 | 1 | 60 |
| | 300_CLOSE | 2 | 2 | 2 | 0 | 23525 |
| | 310_DISPLAY | 1 | 1 | 1 | 0 | 388 |
| | 700_ADDFIL_ERROR_CHECK | 1 | 1 | 1 | 0 | 9925 |
| | 700_READ_REQUEST_FILE | 2 | 2 | 2 | 1 | 86 |

# Raw Product Measures for New and Changed Code by Program

| Program Name | Manhours | Affected Statements | Pre-Number of Statements | Post-Number of Statements | Pre-Cyclomatic Complexity | Post-Cyclomatic Complexity | Pre-Information Flow | Post-Information Flow |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| **Changed Code** | | | | | | | | |
| | | | | | | | | |
| NXR7WQ | 8 | 2 | 36 | 38 | 12 | 12 | 12960 | 16924 |
| NXICWQ | 1 | 5 | 374 | 373 | 125 | 124 | 45827835 | 49728430 |
| NXIDWQ | 4 | 5 | 435 | 431 | 183 | 182 | 32589413 | 30782629 |
| NXMHWQ | 27 | 53 | 624 | 676 | 184 | 191 | 378116858 | 884734941 |
| NXPMWQ | 1 | 6 | 165 | 171 | 35 | 38 | 12545021 | 12580976 |
| NXRGWQ | 1 | 7 | 383 | 388 | 146 | 149 | 9950499 | 11001384 |
| NXPYWQ | 1 | 6 | 326 | 329 | 102 | 104 | 690654368 | 690728240 |
| NXRPWQ | 1 | 7 | 146 | 151 | 61 | 64 | 789107 | 848107 |
| NXU1WQ | 1 | 10 | 684 | 694 | 320 | 323 | 153777927 | 154803863 |
| YG15WQ | 1 | 1 | 519 | 519 | 195 | 195 | 54193695 | 53612520 |
| YGABWQ | 5 | 27 | 306 | 300 | 80 | 82 | 113744450 | 110554252 |
| YGALWQ | 10 | 66 | 669 | 723 | 206 | 216 | 365044311 | 451059644 |
| YGLFWQ | 10 | 38 | 676 | 690 | 210 | 214 | 287908338 | 300004583 |
| YGOBWQ | 5 | 12 | 532 | 535 | 211 | 214 | 2767000000 | 2767000000 |
| YGORWQ | 5 | 3 | 303 | 300 | 100 | 99 | 2694783 | 2651247 |
| YGPFWQ | 5 | 12 | 171 | 182 | 61 | 63 | 5118565 | 5676940 |
| NXMBWQ | 20 | 15 | 285 | 300 | 69 | 82 | 46975611 | 50182077 |
| YGDRWQ | 8 | 7 | 376 | 377 | 88 | 88 | 241613163 | 246707177 |
| YGCTWQ | 250 | 356 | 287 | 643 | 69 | 170 | 10503955 | 32935457 |
| YGUCWQ | 10 | 6 | 493 | 499 | 140 | 141 | 1913000000 | 2097000000 |
| YGDGWQ | 2 | 2 | 33 | 33 | 8 | 8 | 285894 | 285894 |
| YGRNWQ | 96 | 48 | 286 | 341 | 160 | 187 | 11425515 | 29562292 |
| YGADWQ | 40 | 54 | 416 | 466 | 169 | 186 | 5288988 | 10758068 |
| YGEDWQ | 10 | 64 | 849 | 898 | 514 | 521 | 20791611 | 28126833 |
| YGBTWQ | 5 | 1 | 151 | 152 | 14 | 14 | 4396808 | 4396808 |
| YGOTWQ | 5 | 1 | 110 | 111 | 40 | 40 | 3164862 | 3171486 |
| NXM1WQ | 5 | 12 | 51 | 63 | 27 | 37 | 11300 | 19668 |
| NXMPWQ | 5 | 26 | 421 | 447 | 156 | 179 | 54997739 | 61681135 |
| | | | | | | | | |
| **New Code** | | | | | | | | |
| | | | | | | | | |
| NXD0WQ | 8 | 12 | | 12 | | 7 | | 1152 |
| NXD1WQ | 8 | 24 | | 24 | | 12 | | 2626 |
| NXD2WQ | 8 | 52 | | 52 | | 21 | | 41292 |
| NXD3WQ | 8 | 25 | | 25 | | 12 | | 3072 |

## Raw Product Measures for New and Changed Code by Program

| Program Name | Manhours | Affected Statements | Pre-Number of Statements | Post-Number of Statements | Pre-Cyclomatic Complexity | Post-Cyclomatic Complexity | Pre-Information Flow | Post-Information Flow |
|---|---|---|---|---|---|---|---|---|
| NXD4WQ | 8 | 12 | | 12 | | 6 | | 1530 |
| NXD5WQ | 8 | 12 | | 12 | | 6 | | 1530 |
| NXD7WQ | 8 | 12 | | 12 | | 7 | | 1152 |
| NXIEWQ | 175 | 220 | | 220 | | 81 | | 46292137 |
| YGCTWQ | 50 | 356 | | 356 | | 101 | | 22431502 |
| NXIFWQ | 66 | 116 | | 116 | | 47 | | 15456570 |

## *Appendix H*

Scatter Plots of Compared Measures

List of Figures

# Initial Manhour Data

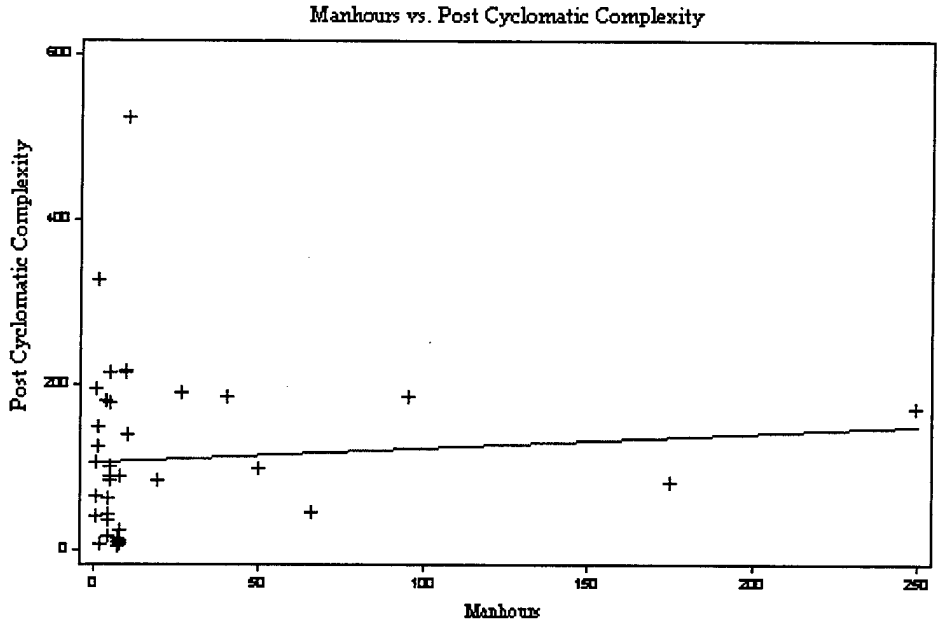



**Figure H-1.** Manhours vs. Post-Cyclomatic Complexity

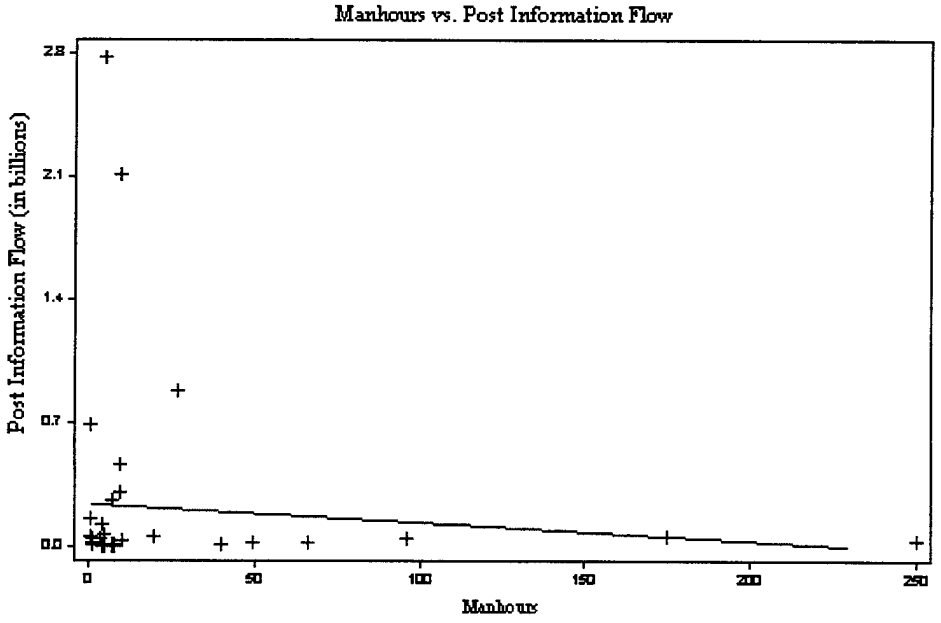



**Figure H-2.** Manhours vs. Post-Information Flow

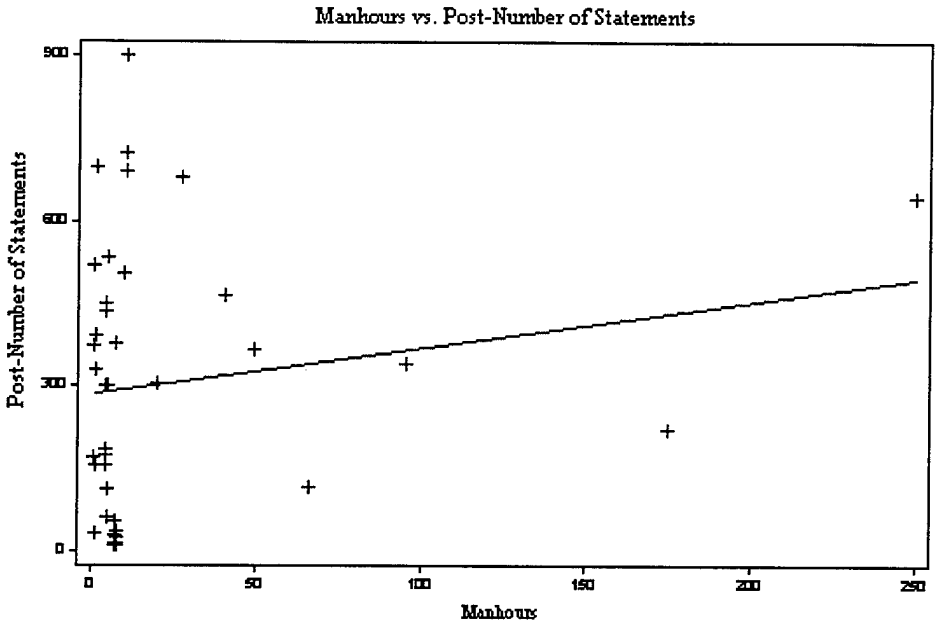



**Figure H-3.** Manhours vs. Post-Number of Statements

## Manhour Data (Changed Code)

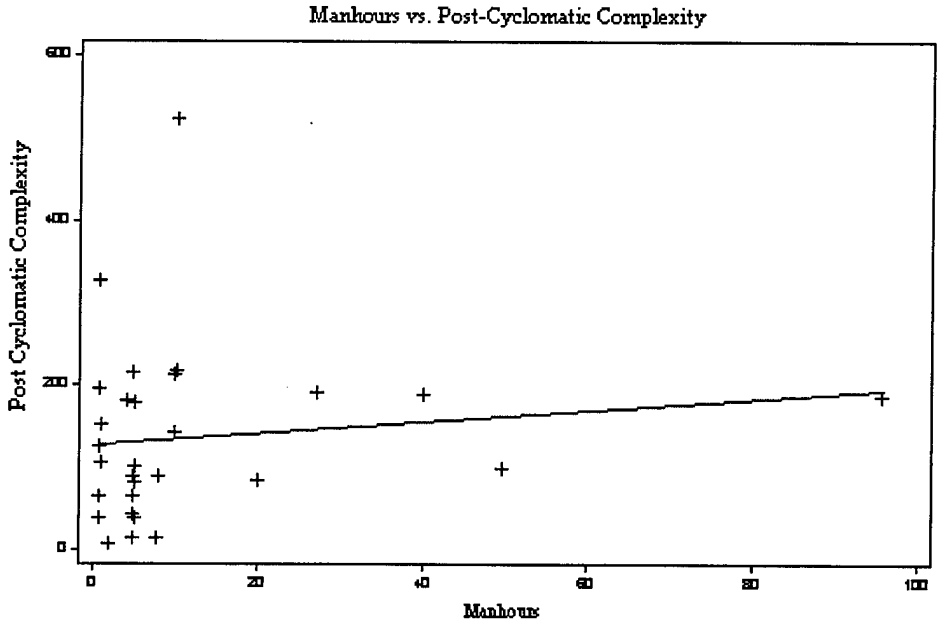



**Figure H-4.** Manhours vs. Post-Cyclomatic Complexity

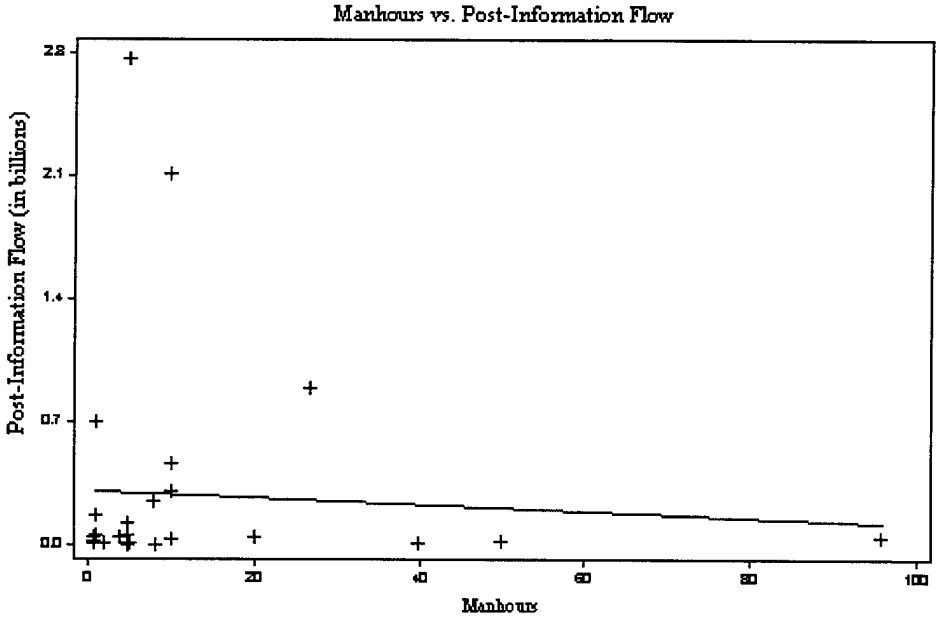



**Figure H-5.** Manhours vs. Post-Information Flow

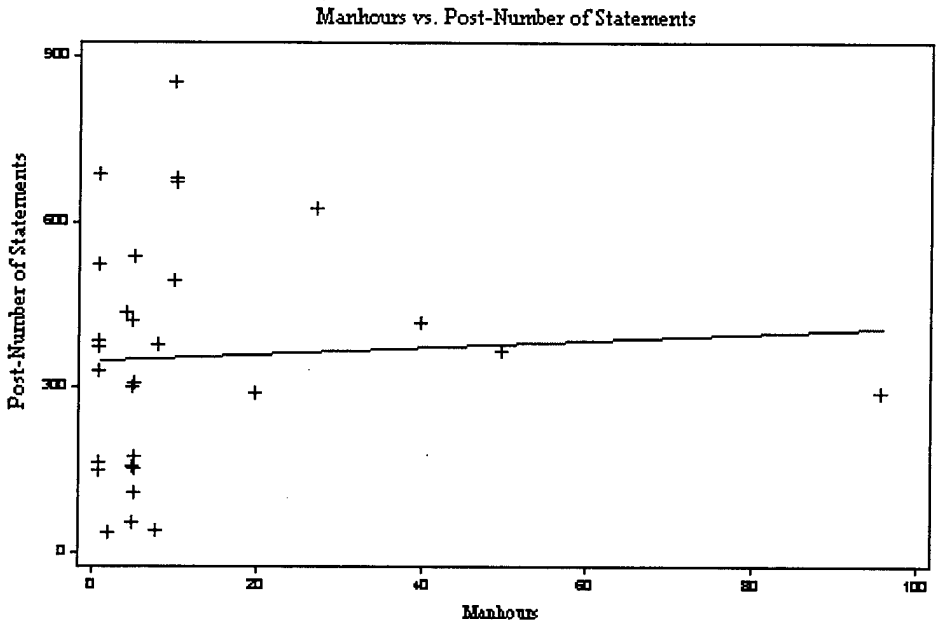



**Figure H-6.** Manhours vs. Post-Number of Statements

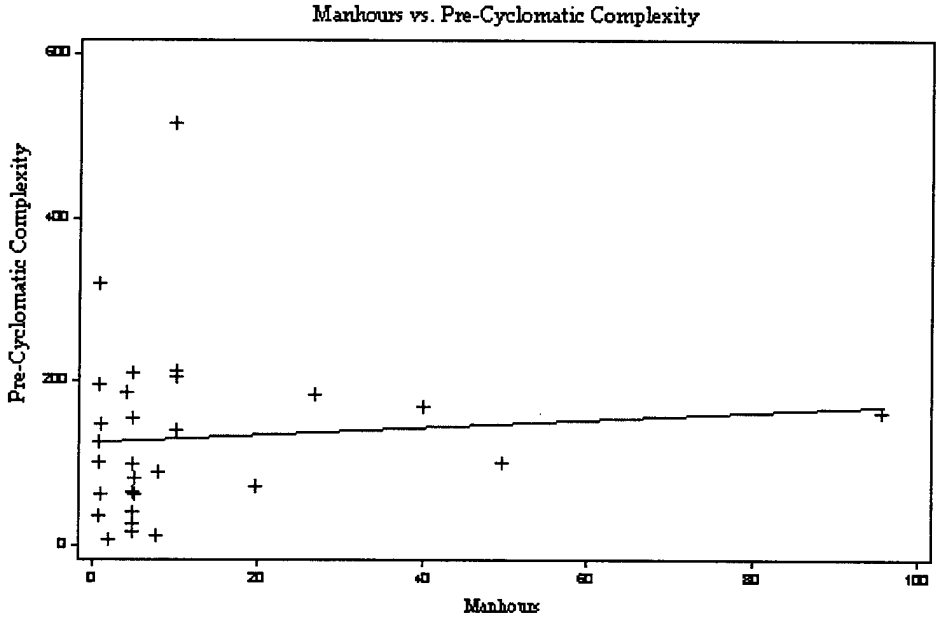



**Figure H-7.** Manhours vs. Pre-Cyclomatic Complexity

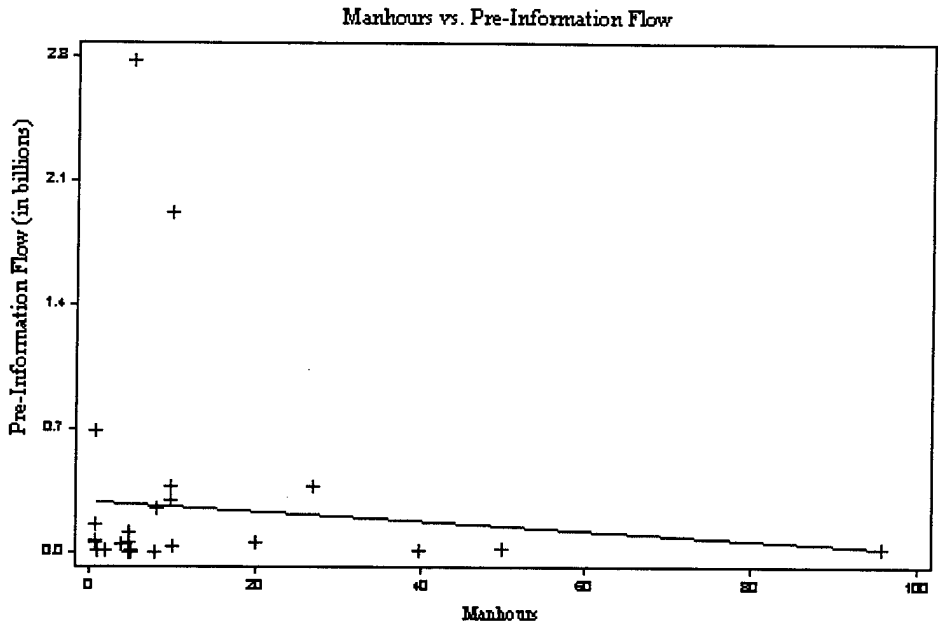



**Figure H-8.** Manhours vs. Pre-Information Flow

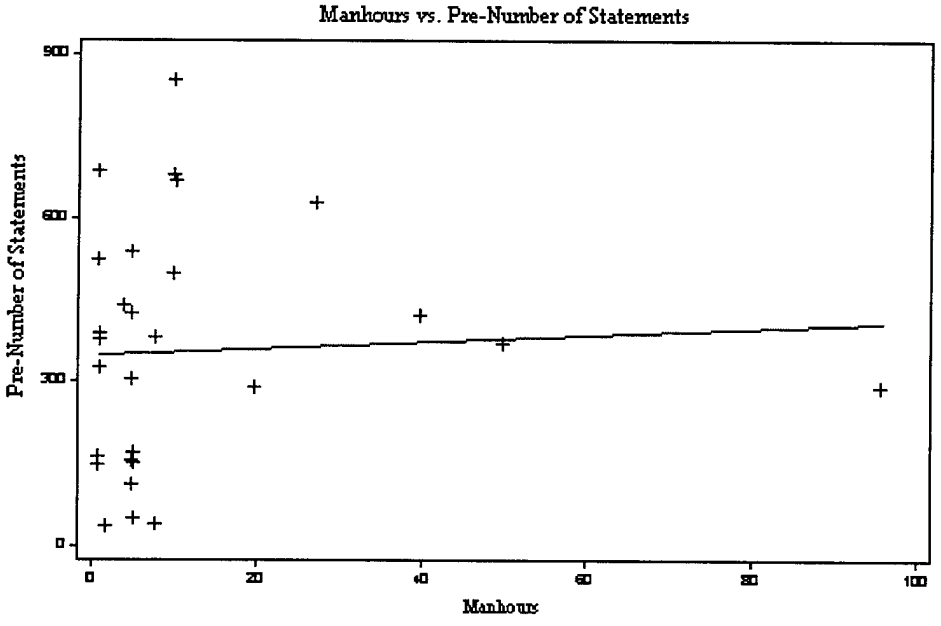



**Figure H-9.** Manhours vs. Pre-Number of Statements

## Product Measures vs. Product Measures

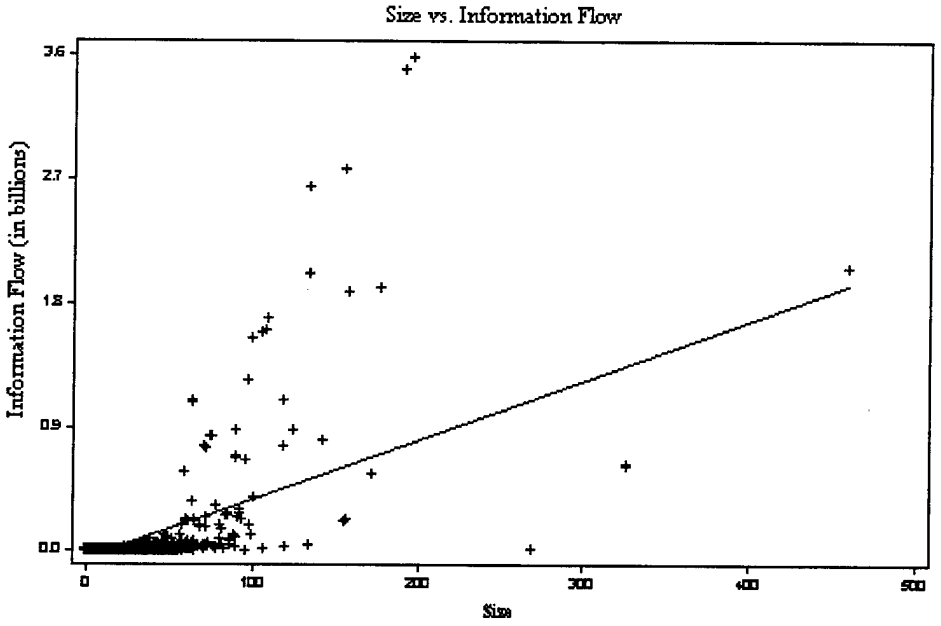



**Figure H-10.** Size vs. Information Flow

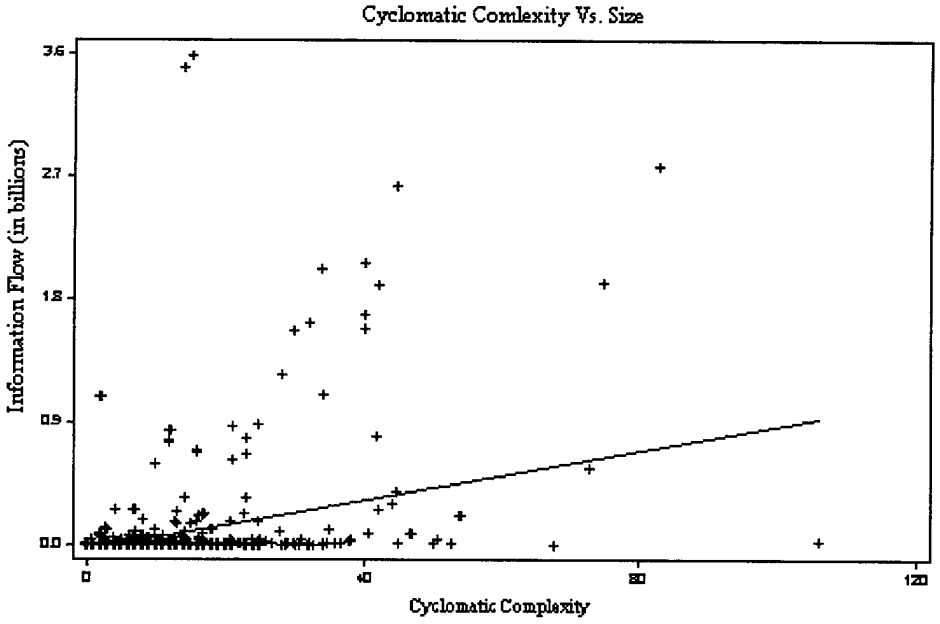



**Figure H-11.** Cyclomatic Complexity vs. Information Flow

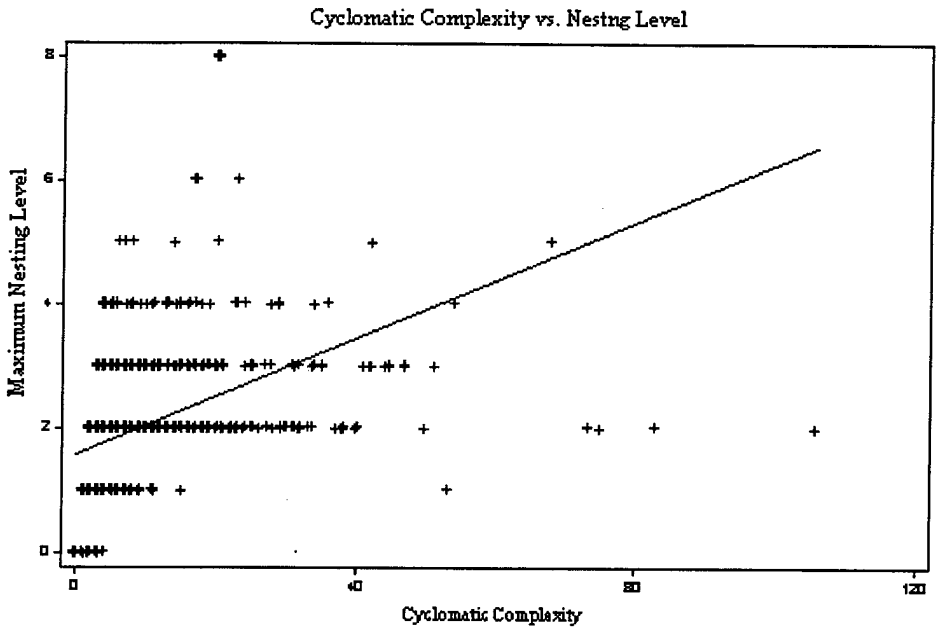



**Figure H-12.** Cyclomatic Complexity vs. Maximum Nesting Level

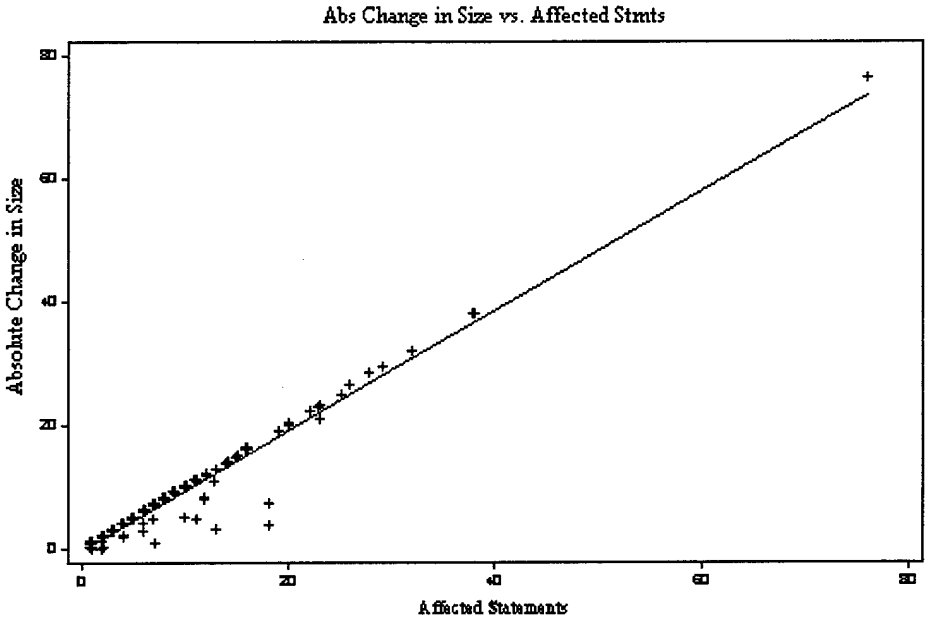



**Figure H-13.** Affected Statements vs. Abs Change in Size

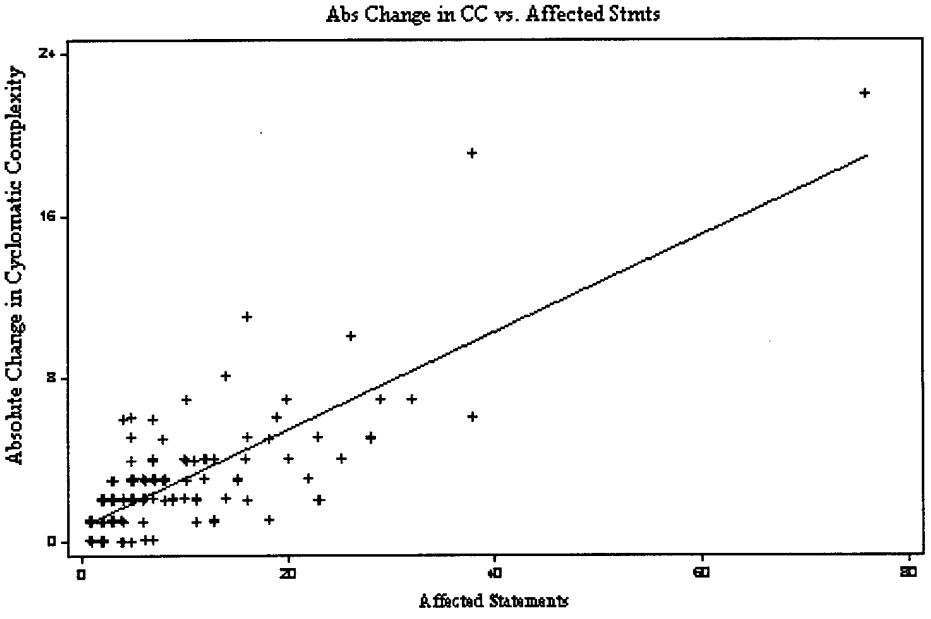



**Figure H-14.** Affected Statements vs. Abs Change in Cyclomatic Complexity

| REPORT DOCUMENTATION PAGE | Form Approved OMB No. 0704-0188 |
|---|---|



| 1. AGENCY USE ONLY *(Leave blank)* | 2. REPORT DATE | 3. REPORT TYPE AND DATES COVERED |
|---|---|---|
| | 21 Nov 97 | Master's Thesis |

**4. TITLE AND SUBTITLE**
An Investigation Into the Use of Software Metrics for Cobol Systems

**5. FUNDING NUMBERS**

**6. AUTHOR(S)**
Richard E. Boone, 1Lt, USAF

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**
Air Force Institue of Technology,
WPAFB OH 45433-7765

**8. PERFORMING ORGANIZATION REPORT NUMBER**
AFIT/GCS/ENG/97D-3

**9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)**
HQ AMC/CSS
SCOTT AFB IL

**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**

**11. SUPPLEMENTARY NOTES**

**12a. DISTRIBUTION AVAILABILITY STATEMENT**
Approved for public release; distribution unlimited

**12b. DISTRIBUTION CODE**

**13. ABSTRACT** *(Maximum 200 words)*

This thesis investigated several hypotheses that specific product measures could be used to predict later software lifecycle process or product measures. It collected software product and process measures from four consecutive major releases of a large Cobol legacy system (400K LOC). The types of product measures used were size and specific complexity measures.

A statistical software package was used to calculate sample correlation coefficients between the measures. A 95% confidence interval was computed for each sample correlation coefficient that showed a strong or moderate linear correlation. The maintenance process measures provided were manhours used for each program changed or added, and defects detected during each change request. Sample correlation coefficients were derived to see if product measures such as size and complexity could reveal trends that could be used to estimate other software lifecycle measures such as effort or defects.

The hypotheses to this research could neither be accepted nor rejected because the process measures collected by the system's owners were recorded at a level too high for sound statistical analysis. Weaknesses are identified in the way these process measures are collected, and suggestions are provided on how process measures can be better identified and recorded.

**14. SUBJECT TERMS**
Software Engineering, Software Metrics, COBOL, Software Measures, Refine COBOL

**15. NUMBER OF PAGES**

**16. PRICE CODE**

| 17. SECURITY CLASSIFICATION OF REPORT | 18. SECURITY CLASSIFICATION OF THIS PAGE | 19. SECURITY CLASSIFICATION OF ABSTRACT | 20. LIMITATION OF ABSTRACT |
|---|---|---|---|
| Unclassified | Unclassified | Unclassified | UL |